

Oracle[®] Secure Global Desktop

Administration Guide for Version 4.6

Copyright © 2010, 2011, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related software documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. UNIX is a registered trademark licensed through X/Open Company, Ltd.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Copyright © 2010, 2011, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf disposition de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique :

U.S. GOVERNMENT RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. UNIX est une marque déposée concédée sous licence par X/Open Company, Ltd.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation.



Adobe PostScript

Contents

Preface xxxv

1. Networking and Security 1

Overview of Networks and Security 1

Connections Between Client Devices and SGD Servers 2

Connections Between SGD Servers and Application Servers 2

Connections Between SGD Servers in an Array 3

DNS Names 3

Configuring External DNS Names 5

Changing the Peer DNS Name of an SGD Server 6

Proxy Servers 8

Supported Proxy Servers 8

Configuring Client Proxy Settings 9

Proxy Server Timeouts 11

Configuring Server-Side Proxy Servers 11

Firewalls 13

Firewalls Between Client Devices and SGD Servers 13

Firewalls Between SGD Servers 14

Firewalls Between SGD Servers and Application Servers 15

Other Firewalls 17

| | |
|---|-----------|
| Secure Connections to SGD Servers | 18 |
| SSL Certificates | 19 |
| Firewall Traversal | 24 |
| Enabling Secure Connections (Automatic Configuration) | 25 |
| Enabling Secure Connections (Manual Configuration) | 28 |
| Secure Connections and Security Warnings | 33 |
| Tuning Secure Connections to SGD Servers | 37 |
| Tuning the SSL Daemon | 38 |
| Using External SSL Accelerators | 40 |
| Selecting a Cipher Suite for Secure Connections | 41 |
| Using Connection Definitions | 43 |
| 2. User Authentication | 47 |
| Secure Global Desktop Authentication | 47 |
| User Identity | 48 |
| User Profile | 49 |
| System Authentication Mechanisms | 49 |
| Password Expiry | 50 |
| Security and Passwords | 51 |
| Active Directory Authentication | 52 |
| How Active Directory Authentication Works | 52 |
| Setting Up Active Directory Authentication | 53 |
| Preparing for Active Directory Authentication | 54 |
| Configuring SGD for Kerberos Authentication | 56 |
| ▼ How to Enable Active Directory Authentication | 60 |
| Anonymous User Authentication | 61 |
| How Anonymous User Authentication Works | 62 |
| ▼ How to Enable Anonymous User Authentication | 63 |
| LDAP Authentication | 63 |

| | |
|---|----|
| How LDAP Authentication Works | 63 |
| Setting Up LDAP Authentication | 65 |
| Preparing for LDAP Authentication | 65 |
| ▼ How to Enable LDAP Authentication | 67 |
| SecurID Authentication | 69 |
| How SecurID Authentication Works | 69 |
| Setting Up SecurID Authentication | 70 |
| Configuring SGD Servers as Agent Hosts | 71 |
| ▼ How to Enable SecurID Authentication | 72 |
| Third-Party Authentication | 72 |
| How Third-Party Authentication Works | 73 |
| Setting Up Third-Party Authentication | 75 |
| ▼ How to Enable Third-Party Authentication | 76 |
| SGD Administrators and Third-Party Authentication | 77 |
| Trusted Users and Third-Party Authentication | 77 |
| UNIX System Authentication | 80 |
| How UNIX System Authentication Works | 80 |
| UNIX System Authentication and PAM | 82 |
| ▼ How to Enable UNIX System Authentication | 83 |
| Windows Domain Authentication | 83 |
| How Windows Domain Authentication Works | 84 |
| ▼ How to Enable Windows Domain Authentication | 85 |
| Passwords, Domains, and Domain Controllers | 85 |
| Tuning Directory Services for Authentication | 86 |
| Filtering LDAP or Active Directory Logins | 87 |
| LDAP Discovery Timeout | 90 |
| Using Service Objects | 90 |
| Password Expiry | 95 |

| | |
|--|------------|
| LDAP Password Update Mode | 96 |
| Sites | 96 |
| Whitelists | 97 |
| Blacklists | 97 |
| Search Only the Global Catalog | 97 |
| Suffix Mappings | 98 |
| Domain Lists | 99 |
| Lookup Cache Timeout | 99 |
| LDAP Operation Timeout | 99 |
| Active Directory Authentication and LDAP Discovery | 100 |
| Troubleshooting Secure Global Desktop Authentication | 102 |
| Setting Log Filters for Authentication Problems | 102 |
| Denying Users Access to SGD After Failed Login Attempts | 103 |
| Users Cannot Log In to Any SGD Server | 104 |
| Using Shared Accounts for Guest Users | 105 |
| Solaris OS Users Cannot Log in When Security is Enabled | 105 |
| An Ambiguous User Name Dialog Is Displayed When a User Tries to Log in | 106 |
| 3. Publishing Applications to Users | 107 |
| Organizations and Objects | 107 |
| Organizational Hierarchies | 108 |
| SGD Object Types | 110 |
| Designing the Organizational Hierarchy | 115 |
| Naming Objects in the Organizational Hierarchy | 116 |
| Populating the SGD Organizational Hierarchy Using a Batch Script | 117 |
| LDAP Mirroring | 118 |
| SGD Administrators | 122 |
| Publishing Applications | 124 |

| | |
|--|------------|
| Local Assignments | 125 |
| LDAP Assignments | 127 |
| Reviewing Assignments | 132 |
| Tuning LDAP Group Searches | 132 |
| Managing the Directory Services Cache | 134 |
| Troubleshooting LDAP Assignments | 135 |
| 4. Configuring Applications | 137 |
| Windows Applications | 137 |
| Configuring Windows Application Objects | 138 |
| Creating Windows Application Objects on the Command Line | 140 |
| Configuring Microsoft Windows Terminal Services for Use With SGD | 140 |
| Licensing Microsoft Windows Terminal Services | 144 |
| Microsoft Windows Remote Desktop | 145 |
| Seamless Windows | 146 |
| Key Handling for Windows Terminal Services | 147 |
| Returning Client Device Information for Windows Terminal Services Sessions | 149 |
| The SGD Remote Desktop Client | 149 |
| Running Windows Applications on Client Devices | 153 |
| X Applications | 153 |
| Configuring X Application Objects | 153 |
| Supported X Extensions | 155 |
| X Authorization | 155 |
| X Fonts | 156 |
| Keyboard Maps | 158 |
| Character Applications | 159 |
| Configuring Character Application Objects | 159 |
| Terminal Emulator Keyboard Maps | 161 |

| | |
|---|-----|
| Terminal Emulator Attribute Maps | 166 |
| Terminal Emulator Color Maps | 167 |
| Dynamic Launch | 169 |
| Dynamic Application Servers | 169 |
| Dynamic Applications | 172 |
| Client Overrides | 174 |
| Using My Desktop | 175 |
| Integrating SGD With Oracle VDI | 176 |
| Using SSH | 178 |
| SSH Support | 179 |
| Configuring the SSH Client | 179 |
| Enabling X11 Forwarding for X Applications | 181 |
| Using SSH and the X Security Extension | 182 |
| Using SSH and X Authorization | 182 |
| Using Advanced SSH Functions | 183 |
| Application Authentication | 183 |
| Login Scripts | 184 |
| Configuring Application Authentication | 185 |
| The Application Server Password Cache | 185 |
| Input Methods and UNIX Platform Applications | 188 |
| Adding Support for System Prompts in Different Languages | 189 |
| Using RSA SecurID for Application Authentication | 189 |
| Tips on Configuring Applications | 189 |
| Starting an Application or Desktop Session Without Displaying a Webtop | 190 |
| Using Multihead Or Dual Head Monitors | 191 |
| Improving the Performance of Windows Applications | 194 |
| Improving the Performance of Java Desktop System Desktop Sessions or Applications | 195 |

| | |
|--|------------|
| Documents and Web Applications | 196 |
| Creating a Virtual Classroom | 196 |
| Configuring Common Desktop Environment Applications | 198 |
| Configuring VMS Applications | 201 |
| 3270 and 5250 Applications | 202 |
| Troubleshooting Applications | 202 |
| An Application Does Not Start | 203 |
| An Application Exits Immediately After Starting | 207 |
| Applications Fail To Start When X Authorization Is Enabled | 207 |
| Applications Disappear After About Two Minutes | 209 |
| An Application Session Does Not End When the User Exits an Application | 210 |
| Users Can Start Applications With Different User Names and Passwords | 212 |
| Using Windows Terminal Services, Users Are Prompted for User Names and Passwords Too Often | 213 |
| Using Shadowing to Troubleshoot a User's Problem | 214 |
| A Kiosk Application Is Not Appearing Full-Screen | 215 |
| An Application's Animation Appears 'Jumpy' | 215 |
| Font Problems with X Applications | 215 |
| Display Problems With High Color X Applications | 216 |
| Clipped Windows With Client Window Management Applications | 218 |
| Emulating a Sun Keyboard | 218 |
| Display Update Issues When Shadowing Over a Low Bandwidth Connection | 220 |
| Troubleshooting Mouse Drag Delay Issues | 220 |
| Incorrect Time Zone Name Shown in Windows Applications | 221 |
| 5. Client Device Support | 223 |
| Printing | 223 |
| Overview of SGD Printing | 224 |

| | |
|---|-----|
| Setting Up Printing | 225 |
| Configuring Microsoft Windows Application Servers for Printing | 225 |
| Configuring UNIX and Linux Platform Application Servers for Printing | 228 |
| Configuring an SGD Server for Printing | 233 |
| Configuring Printing to Microsoft Windows Client Devices | 237 |
| Configuring Printing to UNIX, Linux, and Mac OS X Platform Client Devices | 241 |
| Managing Printing | 244 |
| Users Cannot Print From Applications Displayed Through SGD | 246 |
| Troubleshooting Other Printing Problems | 254 |
| Client Drive Mapping | 258 |
| Setting Up Client Drive Mapping | 259 |
| Configuring UNIX and Linux Platform Application Servers for CDM | 259 |
| Configuring an NFS Share for CDM | 260 |
| Starting CDM Processes on the Application Server | 262 |
| Configuring Microsoft Windows Application Servers for CDM | 262 |
| Enabling CDM Services in SGD | 262 |
| Running UNIX Platform CDM With Another SMB Service | 263 |
| Configuring the Client Drives Available to Users | 264 |
| Troubleshooting Client Drive Mapping | 267 |
| Logging for CDM | 275 |
| Audio | 276 |
| Setting Up Audio | 276 |
| Configuring Microsoft Windows Application Servers for Audio | 277 |
| Configuring UNIX and Linux Platform Application Servers for Audio | 277 |
| Configuring X Applications for Audio | 279 |
| Enabling SGD Audio Services | 279 |
| Configuring Client Devices for Audio | 280 |
| Troubleshooting Audio in Applications | 281 |

| | |
|---|------------|
| Copy and Paste | 287 |
| Using Copy and Paste | 287 |
| Controlling Copy and Paste in Applications | 288 |
| An Example of Using Clipboard Security Levels | 289 |
| Tips on Configuring Copy and Paste | 290 |
| Copy and Paste Troubleshooting | 291 |
| Smart Cards | 292 |
| Using Smart Cards With Windows Applications | 292 |
| Setting Up Access to Smart Cards | 293 |
| Configuring the Microsoft Windows Application Server for Smart Cards | 293 |
| Enabling Smart Cards in SGD | 294 |
| Configuring Smart Card Readers on Client Devices | 295 |
| ▼ How to Log In to a Microsoft Windows Application Server With a Smart Card | 296 |
| Troubleshooting Smart Cards | 296 |
| Serial Ports | 298 |
| Setting Up Access to Serial Ports | 298 |
| Configuring the Microsoft Windows Application Server | 298 |
| Enabling Serial Port Access in SGD | 298 |
| Configuring the Client Device | 299 |
| 6. SGD Client and Webtop | 301 |
| The SGD Client | 301 |
| Overview of the SGD Client | 301 |
| Installing the SGD Client | 303 |
| Automatic Installation of the SGD Client | 303 |
| ▼ How to Enable Automatic Installation for Roaming User Profiles | 304 |
| Manual Installation of the SGD Client | 305 |
| Running the SGD Client From the Command Line | 306 |

| | |
|---|------------|
| Using SGD Without Java Technology | 309 |
| Client Profiles | 311 |
| Client Profiles and the SGD Client | 312 |
| Managing Client Profiles | 312 |
| ▼ How to Configure Client Profile Editing for Users | 313 |
| Client Profile Settings | 315 |
| About the Profile Cache | 317 |
| Microsoft Windows Users With Roaming User Profiles | 319 |
| Integrated Mode | 320 |
| Working in Integrated Mode | 321 |
| Setting Up the SGD Client for Integrated Mode | 322 |
| Authentication Token Authentication | 323 |
| Configuring the Client Profile for Integrated Mode | 328 |
| Configuring Applications for Integrated Mode | 329 |
| Webtops | 329 |
| Setting the Language for the Webtop | 329 |
| 7. SGD Servers, Arrays, and Load Balancing | 333 |
| Arrays | 333 |
| The Structure of an Array | 334 |
| Replicating Data Across the Array | 334 |
| Communication Between Array Members | 335 |
| Secure Intra-Array Communication | 336 |
| Managing Arrays and SGD Servers | 338 |
| Array Resilience | 338 |
| Configuring Arrays | 343 |
| Configuring Array Resilience | 348 |
| Load Balancing | 352 |
| User Session Load Balancing | 353 |

| | |
|--|-----|
| Application Session Load Balancing | 361 |
| Application Load Balancing | 362 |
| Load Balancing Groups | 363 |
| How Application Load Balancing Works | 364 |
| How Advanced Load Management Works | 371 |
| Tuning Application Load Balancing | 372 |
| Editing Application Load Balancing Properties | 375 |
| SGD Web Server and Administration Console | 379 |
| Introducing the SGD Web Server | 380 |
| Securing the SGD Web Server | 380 |
| Using the Administration Console | 381 |
| Administration Console Configuration Settings | 383 |
| Securing Access to the Administration Console | 385 |
| Monitoring and Logging | 386 |
| The SGD Datastore | 386 |
| User Sessions and Application Sessions | 387 |
| Using Log Filters to Troubleshoot Problems With an SGD Server | 391 |
| Using Log Filters for Auditing | 398 |
| Using Log Filters to Troubleshoot Problems With Protocol Engines | 402 |
| SGD Web Server Logging | 406 |
| SGD Client Logging | 407 |
| SGD Server Certificate Stores | 408 |
| The CA Certificate Truststore | 408 |
| The Client Certificate Store | 409 |
| SGD Installations | 411 |
| About Your SGD Installation | 411 |
| Backing Up and Restoring an SGD Installation | 414 |
| Troubleshooting Arrays and Load Balancing | 419 |

| | |
|---|------------|
| Troubleshooting Array Resilience | 420 |
| Troubleshooting Clock Synchronization Issues | 422 |
| Troubleshooting Advanced Load Management | 423 |
| SGD Uses Too Much Network Bandwidth | 427 |
| Users Cannot Connect to an SGD Server When It Is In Firewall Traversal Mode | 428 |
| Users Cannot Relocate Their Sessions | 429 |
| A. Global Settings and Caches | 431 |
| Secure Global Desktop Authentication Tab | 432 |
| The Authentication Wizard | 432 |
| Token Generation | 434 |
| Password Cache | 435 |
| Third-Party Authentication | 435 |
| System Authentication | 436 |
| Search Local Repository | 436 |
| Search LDAP Repository | 437 |
| Use Default Third-Party Identity | 437 |
| Use Default LDAP Profile | 438 |
| Use Closest Matching LDAP Profile | 439 |
| LDAP/Active Directory | 440 |
| Unix | 440 |
| Authentication Token | 441 |
| Windows Domain Controller | 441 |
| SecurID | 442 |
| Anonymous | 442 |
| Search Unix User ID in Local Repository | 443 |
| Search Unix Group ID in Local Repository | 443 |
| Use Default User Profile | 444 |

| | |
|---------------------------------------|-----|
| Windows Domain | 444 |
| Active Directory | 445 |
| LDAP | 445 |
| Service Objects Tab | 446 |
| The Service Objects List Table | 446 |
| Name | 447 |
| Type | 448 |
| Enabled | 448 |
| URLs | 448 |
| User Name and Password | 449 |
| Connection Security | 450 |
| Active Directory Base Domain | 450 |
| Active Directory Default Domain | 450 |
| Application Authentication Tab | 451 |
| Password Cache Usage | 451 |
| Action When Password Expired | 452 |
| Smart Card Authentication | 453 |
| Dialog Display | 453 |
| “Save Password” Box | 454 |
| “Always Use Smart Card” Box | 455 |
| Display Delay | 456 |
| “Launch Details” Pane | 456 |
| Communication Tab | 457 |
| Unencrypted Connections Port | 457 |
| Encrypted Connections Port | 458 |
| AIP Keepalive Frequency | 459 |
| Timeout for User Session Resumability | 459 |
| Timeout for General Resumability | 460 |

| | |
|--|-----|
| Resource Synchronization Service | 461 |
| User Session Idle Timeout | 462 |
| Performance Tab | 462 |
| Application Session Load Balancing | 463 |
| Application Load Balancing | 464 |
| Client Device Tab | 465 |
| Windows Client Drive Mapping | 465 |
| Unix Client Drive Mapping | 466 |
| Dynamic Drive Mapping | 467 |
| Windows Audio | 467 |
| Windows Audio Sound Quality | 468 |
| Unix Audio | 469 |
| Unix Audio Sound Quality | 469 |
| Smart Card | 470 |
| Serial Port Mapping | 470 |
| Copy and Paste | 471 |
| Client's Clipboard Security Level | 472 |
| Time Zone Map File | 472 |
| Editing | 473 |
| Printing Tab | 474 |
| Client Printing | 474 |
| Universal PDF Printer | 475 |
| Make Universal PDF Printer the Default | 475 |
| Universal PDF Viewer | 476 |
| Make Universal PDF Viewer the Default | 477 |
| Postscript Printer Driver | 477 |
| Security Tab | 478 |
| New Password Encryption Key | 478 |

| | |
|---|------------|
| Timeout for Print Name Mapping | 479 |
| Connection Definitions | 480 |
| X Authorization for X Display | 481 |
| Monitoring Tab | 481 |
| Log Filter | 482 |
| Billing Service | 482 |
| Resilience Tab | 483 |
| Array Failover | 483 |
| Monitor Interval | 484 |
| Monitor Attempts | 484 |
| Find Primary Interval | 485 |
| Find Primary Attempts | 486 |
| Action When Failover Ends | 486 |
| Backup Primaries | 487 |
| Caches Tab | 488 |
| Passwords Tab | 488 |
| Description | 488 |
| Command Line | 490 |
| Tokens Tab | 490 |
| Description | 490 |
| Command Line | 491 |
| B. Secure Global Desktop Server Settings | 493 |
| Secure Global Desktop Servers Tab | 494 |
| The Secure Global Desktop Server List Table | 494 |
| General Tab | 495 |
| External DNS Names | 495 |
| User Login | 496 |
| Redirection URL | 497 |

| | |
|-------------------------------------|-----|
| Security Tab | 498 |
| Connection Types | 498 |
| SSL Accelerator Support | 498 |
| Firewall Forwarding URL | 499 |
| Performance Tab | 500 |
| Maximum Simultaneous Requests | 500 |
| Maximum Simultaneous User Sessions | 500 |
| Maximum File Descriptors | 501 |
| JVM Size | 502 |
| Daily Resource Synchronization Time | 503 |
| Load Balancing Groups | 503 |
| Protocol Engines Tab | 504 |
| Character Protocol Engine Tab | 505 |
| Maximum Sessions | 505 |
| Exit Timeout | 505 |
| Command-Line Arguments | 506 |
| X Protocol Engine Tab | 507 |
| Monitor Resolution | 507 |
| Font Path | 507 |
| RGB Database | 508 |
| Keyboard Map | 509 |
| Client Window Size | 509 |
| Session Start Timeout | 510 |
| Maximum Sessions | 511 |
| Exit Timeout | 511 |
| Command-Line Arguments | 512 |
| Execution Protocol Engine Tab | 512 |
| Maximum Sessions | 512 |

| | |
|--|------------|
| Exit Timeout | 513 |
| Login Script Directory | 514 |
| Command-Line Arguments | 514 |
| Channel Protocol Engine Tab | 515 |
| Packet Compression | 515 |
| Packet Compression Threshold | 516 |
| Exit Timeout | 516 |
| Print Protocol Engine Tab | 517 |
| Packet Compression | 517 |
| Packet Compression Threshold | 517 |
| Exit Timeout | 518 |
| Audio Protocol Engine Tab | 518 |
| Packet Compression | 519 |
| IO Protocol Engine Tab | 519 |
| Packet Compression | 519 |
| User Sessions Tab | 520 |
| The User Session List Table | 520 |
| Application Sessions Tab | 521 |
| The Application Session List Table | 522 |
| C. User Profiles, Applications, and Application Servers | 525 |
| SGD Objects | 525 |
| 3270 Application Object | 526 |
| 5250 Application Object | 528 |
| Application Server Object | 530 |
| Character Application Object | 531 |
| Directory: Organization Object | 533 |
| Directory: Organizational Unit Object | 534 |
| Directory (Light): Active Directory Container Object | 535 |

| | |
|--|-----|
| Directory (Light): Domain Component Object | 535 |
| Document Object | 536 |
| Group Object | 536 |
| User Profile Object | 537 |
| Dynamic Application Object | 539 |
| Dynamic Application Server Object | 539 |
| Windows Application Object | 540 |
| X Application Object | 542 |
| Attributes Reference | 545 |
| Address | 545 |
| Allow SSH Downgrade | 546 |
| Answerback Message | 546 |
| Application Command | 547 |
| Application Load Balancing | 548 |
| Application Resumability | 549 |
| Application Resumability: Timeout | 551 |
| Application Sessions Tab | 552 |
| Application Start | 554 |
| Arguments | 554 |
| Arguments for Command | 555 |
| Assigned Applications Tab | 556 |
| Assigned User Profiles Tab | 558 |
| Attribute Map | 563 |
| Audio Redirection Library | 563 |
| Background Color | 564 |
| Bandwidth Limit | 565 |
| Border Style | 566 |
| Client Drive Mapping | 567 |

| | |
|--|-----|
| Client Printing | 569 |
| Client Printing: Override | 570 |
| Client Profile Editing | 572 |
| Code Page | 573 |
| Color Depth | 574 |
| Color Map | 576 |
| Color Quality | 576 |
| Command Compression | 578 |
| Command Execution | 579 |
| Comment | 579 |
| Connection Closed Action | 580 |
| Connection Method | 581 |
| Connections | 582 |
| Connection Method: SSH Arguments | 584 |
| Console Mode | 585 |
| Copy and Paste | 585 |
| Copy and Paste: Application's Clipboard Security Level | 587 |
| Cursor | 588 |
| Cursor Key Codes Modification | 588 |
| Cursor Settings | 589 |
| Cursor Shadow | 589 |
| Delayed Updates | 590 |
| Desktop Wallpaper | 591 |
| Displayed Soft Buttons | 591 |
| Domain Name | 592 |
| Email Address | 593 |
| Emulation Type | 593 |
| Environment Variables | 594 |

- Escape Sequences 595
- Euro Character 595
- 'File' and 'Settings' Menus 596
- Font Family 597
- Font Size 598
- Font Size: Fixed Font Size 598
- Font Smoothing 599
- Foreground Color 599
- Full Window Drag 600
- Graphics Acceleration 601
- Hints 602
- Hosted Applications Tab 602
- Hosting Application Servers Tab 604
- Icon 606
- Inherit Assigned Applications from Parent 607
- Interlaced Images 608
- Keep Launch Connection Open 608
- Keyboard Codes Modification 609
- Keyboard Map 610
- Keyboard Map: Locked 611
- Keyboard Type 612
- Kiosk Mode Escape 612
- Line Wrapping 613
- Load Balancing Groups 614
- Local Client Launch 615
- Login 615
- Login: Multiple 616
- Login Name 617

| | |
|--|-----|
| Login Script | 618 |
| Make Universal PDF Printer the Default | 619 |
| Make Universal PDF Viewer the Default | 620 |
| Mappings Tab | 621 |
| Maximum Count | 622 |
| Members Tab | 622 |
| Menu Animations | 624 |
| Menu Bar | 625 |
| Middle Mouse Timeout | 625 |
| Monitor Resolution | 626 |
| Mouse | 627 |
| Name | 628 |
| Number of Sessions | 630 |
| Numpad Codes Modification | 631 |
| Passwords Tab | 631 |
| Password Cache Usage | 632 |
| Postscript Printer Driver | 633 |
| Printer Preference Caching | 635 |
| Prompt Locale | 635 |
| Remote Audio | 636 |
| Scroll Style | 637 |
| Serial Port Mapping | 637 |
| Server Address | 639 |
| Server Port | 639 |
| Session Termination | 640 |
| SGD Remote Desktop Client | 642 |
| Share Resources Between Similar Sessions | 642 |
| Status Line | 643 |

Surname 644

SWM Local Window Hierarchy 645

Terminal Type 645

Theming 646

Tokens Tab 647

Universal PDF Printer 647

Universal PDF Viewer 648

URL 649

User Assignment 650

User Sessions Tab 652

Virtual Server Broker Class 653

Virtual Server Broker Parameters 654

Window Close Action 654

Window Color 657

Window Color: Custom Color 657

Window Management Keys 658

Window Manager 659

Window Size: Client's Maximum Size 660

Window Size: Columns 661

Window Size: Height 661

Window Size: Lines 662

Window Size: Maximized 662

Window Size: Scale to Fit Window 663

Window Size: Width 664

Window Type 665

Window Type: New Browser Window 668

Working Directory 668

X Security Extension 669

D. Commands 671

The tarantella Command 672

Syntax 672

Description 672

Examples 674

The tarantella archive Command 675

Syntax 675

Description 675

Examples 675

The tarantella array Command 675

Syntax 676

Description 676

Examples 677

tarantella array add_backup_primary 677

tarantella array clean 678

tarantella array detach 679

tarantella array edit_backup_primary 680

tarantella array join 681

tarantella array list 683

tarantella array list_backup primaries 683

tarantella array make_primary 684

tarantella array remove_backup_primary 685

The tarantella cache Command 686

Syntax 687

Description 687

Examples 688

The tarantella config Command 688

Syntax 688

Description 688

Examples 689

tarantella config edit 689

tarantella config list 691

The tarantella emulatorsession Command 692

Syntax 692

Description 692

Examples 693

tarantella emulatorsession list 693

tarantella emulatorsession info 695

tarantella emulatorsession shadow 696

tarantella emulatorsession suspend 697

tarantella emulatorsession end 698

The tarantella help Command 699

Syntax 700

Description 700

Examples 700

The tarantella object Command 700

Syntax 701

Description 701

Examples 703

tarantella object add_host 703

tarantella object add_link 704

tarantella object add_mapping 705

tarantella object add_member 707

tarantella object delete 708

tarantella object edit 709

tarantella object list_attributes 710

| | |
|----------------------------------|-----|
| tarantella object list_contents | 711 |
| tarantella object new_3270app | 712 |
| tarantella object new_5250app | 717 |
| tarantella object new_charapp | 722 |
| tarantella object new_container | 727 |
| tarantella object new_dc | 727 |
| tarantella object new_doc | 728 |
| tarantella object new_dynamicapp | 730 |
| tarantella object new_group | 731 |
| tarantella object new_host | 733 |
| tarantella object new_org | 736 |
| tarantella object new_orgunit | 738 |
| tarantella object new_person | 741 |
| tarantella object new_windowsapp | 744 |
| tarantella object new_xapp | 750 |
| tarantella object remove_host | 755 |
| tarantella object remove_link | 756 |
| tarantella object remove_mapping | 757 |
| tarantella object remove_member | 759 |
| tarantella object rename | 760 |
| tarantella object script | 761 |
| The tarantella passcache Command | 762 |
| Syntax | 762 |
| Description | 762 |
| Examples | 763 |
| tarantella passcache delete | 763 |
| tarantella passcache edit | 765 |
| tarantella passcache list | 767 |

| | |
|--------------------------------|-----|
| tarantella passcache new | 769 |
| The tarantella print Command | 770 |
| Syntax | 771 |
| Description | 771 |
| Examples | 771 |
| tarantella print cancel | 772 |
| tarantella print list | 773 |
| tarantella print move | 774 |
| tarantella print pause | 775 |
| tarantella print resume | 776 |
| tarantella print start | 777 |
| tarantella print status | 778 |
| tarantella print stop | 779 |
| The tarantella query Command | 780 |
| Syntax | 781 |
| Description | 781 |
| Examples | 781 |
| tarantella query audit | 781 |
| tarantella query billing | 784 |
| tarantella query errlog | 786 |
| tarantella query uptime | 787 |
| The tarantella restart Command | 787 |
| Syntax | 788 |
| Description | 788 |
| Examples | 789 |
| tarantella restart sgd | 789 |
| tarantella restart webserver | 790 |
| The tarantella role Command | 791 |

| | |
|---------------------------------|-----|
| Syntax | 792 |
| Description | 792 |
| Examples | 792 |
| tarantella role add_link | 793 |
| tarantella role add_member | 794 |
| tarantella role list | 795 |
| tarantella role list_links | 795 |
| tarantella role list_members | 796 |
| tarantella role remove_link | 797 |
| tarantella role remove_member | 798 |
| The tarantella security Command | 799 |
| Syntax | 800 |
| Description | 800 |
| Examples | 801 |
| tarantella security certinfo | 801 |
| tarantella security certrequest | 803 |
| tarantella security certuse | 805 |
| tarantella security customca | 807 |
| tarantella security decryptkey | 808 |
| tarantella security disable | 809 |
| tarantella security enable | 810 |
| tarantella security fingerprint | 812 |
| tarantella security peerca | 813 |
| tarantella security selfsign | 813 |
| tarantella security start | 814 |
| tarantella security stop | 815 |
| The tarantella service Command | 816 |
| Syntax | 816 |

| | |
|-------------------------------|-----|
| Description | 816 |
| Examples | 816 |
| tarantella service delete | 817 |
| tarantella service edit | 817 |
| tarantella service list | 821 |
| tarantella service new | 822 |
| The tarantella setup Command | 826 |
| Syntax | 826 |
| Description | 826 |
| Examples | 827 |
| The tarantella start Command | 827 |
| Syntax | 827 |
| Description | 827 |
| Examples | 828 |
| tarantella start cdm | 828 |
| tarantella start sgd | 829 |
| tarantella start webserver | 829 |
| The tarantella status Command | 830 |
| Syntax | 831 |
| Description | 831 |
| Examples | 832 |
| The tarantella stop Command | 832 |
| Syntax | 833 |
| Description | 833 |
| Examples | 834 |
| tarantella stop cdm | 834 |
| tarantella stop sgd | 835 |
| tarantella stop webserver | 835 |

The tarantella tokencache Command 836

Syntax 837

Description 837

Examples 837

tarantella tokencache delete 837

tarantella tokencache list 838

The tarantella tscal Command 839

Syntax 840

Description 840

Examples 840

tarantella tscal free 840

tarantella tscal list 842

tarantella tscal return 843

The tarantella uninstall Command 844

Syntax 845

Description 845

Examples 845

The tarantella version Command 845

Syntax 846

Description 846

Examples 846

The tarantella webserver Command 846

Syntax 847

Description 847

Examples 847

tarantella webserver add_trusted_user 847

tarantella webserver delete_trusted_user 848

tarantella webserver list_trusted_users 849

The tarantella webtopsession Command 850

Syntax 850

Description 850

Examples 851

tarantella webtopsession list 851

tarantella webtopsession logout 852

E. Login Scripts 855

Login Scripts Supplied With SGD 855

Login Scripts Used When Configuring Applications 856

Login Scripts Containing Common Code 857

Login Script Tcl Commands and Procedures 858

Controlling the SGD Application Authentication Dialog 858

Controlling the SGD Progress Dialog 862

Controlling the Connection to the Application Server 863

Login Script Variables 866

Guaranteed Login Script Variables 866

Optional Login Script Variables 868

Login Script Timeouts 874

Expect Timeouts 874

Client Timers 875

Other Timeouts 877

Login Script Error Messages 878

F. Third-Party Legal Notices 885

Apache License, Version 2.0 886

Cryptix 892

JACL/TCL 892

OpenMotif 894

| | |
|-----------------------------|------------|
| OpenSSL | 899 |
| TeemTalk | 902 |
| Unicode Character Database | 902 |
| X Window System (X11R6.8.2) | 910 |
| XML Parser | 910 |
| zlib | 911 |
| Glossary | 913 |
| Index | 931 |

Preface

The *Oracle Secure Global Desktop 4.6 Administration Guide* is a comprehensive guide to how to configure, administer, and troubleshoot problems with Oracle Secure Global Desktop (SGD). This document is written for SGD Administrators.

How This Book Is Organized

Chapter 1 describes how to integrate SGD into your network infrastructure and secure the network connections used by SGD.

Chapter 2 describes how users authenticate to an SGD server to log in to SGD. This chapter also covers how users authenticate to an application server to run an application.

Chapter 3 describes how you use organizational hierarchies to manage SGD users and give them access to applications.

Chapter 4 contains advice on configuring applications that users can run through SGD, and how to diagnose and fix problems with applications.

Chapter 5 describes how to enable support for peripherals and other client device features from applications displayed in SGD.

Chapter 6 describes how to install, configure, and run the SGD Client. Webtop configuration is also covered.

Chapter 7 describes how to configure and monitor SGD servers and arrays. Some system administration features of SGD, such as the Administration Console, log filters, and installation backups are also covered.

Appendix A describes global settings which apply to all SGD servers in the array, including the password cache and token cache.

Appendix B describes server settings which apply to the specified SGD server in the array.

Appendix C describes the supported object types in SGD and their attributes. Usage details for setting attributes using the Administration Console are included, along with the equivalent SGD command line.

Appendix D describes the available SGD commands. Examples are included for each command.

Appendix E contains reference information about the SGD login scripts. You can use this information to customize the standard SGD login scripts, or to develop your own login scripts.

Using UNIX Commands

This document might not contain information on basic UNIX® commands and procedures such as shutting down the system, booting the system, and configuring devices. Refer to your system documentation for this information. This document does, however, contain information about specific SGD commands.

Shell Prompts

| Shell | Prompt |
|---------------------------------------|----------------------|
| C shell | <i>machine-name%</i> |
| C shell superuser | <i>machine-name#</i> |
| Bourne shell and Korn shell | \$ |
| Bourne shell and Korn shell superuser | # |

Typographic Conventions

| Typeface | Meaning | Examples |
|------------------|--|---|
| AaBbCc123 | The names of commands, files, and directories; on-screen computer output | Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. % You have mail. |
| AaBbCc123 | What you type, when contrasted with on-screen computer output | % su Password: |
| <i>AaBbCc123</i> | Book titles, new words or terms, words to be emphasized. Replace command-line variables with real names or values. | Read Chapter 6 in the <i>User's Guide</i> . These are called <i>class</i> options. To delete a file, type rm <i>filename</i> . |

Note – Characters display differently depending on browser settings. If characters do not display correctly, change the character encoding in your browser to Unicode UTF-8.

Related Documentation

The following table lists the documentation for this product. The online documentation is available at:

<http://download.oracle.com/docs/cd/E19351-01/index.html>

| Application | Title | Part Number | Format | Location |
|----------------|--|-------------|--------|------------------------|
| Release Notes | <i>Oracle Secure Global Desktop 4.6 Platform Support and Release Notes</i> | 821-1928 | HTML | Online |
| | | | PDF | Software CD and online |
| Installation | <i>Oracle Secure Global Desktop 4.6 Installation Guide</i> | 821-1927 | HTML | Online |
| | | | PDF | Software CD and online |
| Administration | <i>Oracle Secure Global Desktop 4.6 Administration Guide</i> | 821-1926 | HTML | Online |
| | | | PDF | |
| User | <i>Oracle Secure Global Desktop 4.6 User Guide</i> | 821-1925 | HTML | Online |
| | | | PDF | |
| Administration | <i>Oracle Secure Global Desktop 4.6 Gateway Administration Guide</i> | 821-1924 | HTML | Online |
| | | | PDF | |

Third-Party Web Sites

Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused by or in connection with the use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Documentation Feedback

Submit comments about this document to VIRT-DOCS-EXT_WW@ORACLE.COM. Include the title and part number of your document with your feedback:

Oracle Secure Global Desktop 4.6 Administration Guide, part number 821-1926.

Networking and Security

This chapter describes how to integrate Oracle Secure Global Desktop (SGD) into your network infrastructure and secure the network connections used by SGD.

This chapter includes the following topics:

- [“Overview of Networks and Security” on page 1](#)
- [“DNS Names” on page 3](#)
- [“Proxy Servers” on page 8](#)
- [“Firewalls” on page 13](#)
- [“Secure Connections to SGD Servers” on page 18](#)
- [“Tuning Secure Connections to SGD Servers” on page 37](#)

Overview of Networks and Security

When using SGD, client devices never connect directly to application servers. Instead they connect to SGD using Hypertext Transfer Protocol (HTTP) or HTTP over Secure Sockets Layer (HTTPS) and the SGD Adaptive Internet Protocol (AIP). SGD then connects to the application servers on the user’s behalf.

The following are the main network connections involved when using SGD:

- Connections between client devices and SGD servers
- Connections between SGD servers and application servers
- Connections between SGD servers in an array

In a default SGD installation, most network connections are not secure. The following sections describe how you can secure these network connections.

Connections Between Client Devices and SGD Servers

Client devices makes the following connections to SGD servers:

- **HTTP connections.** These are the connections to the SGD web server, used for SGD web services, authentication to SGD, and to display the webtop.
- **AIP connections.** These are the connections between the SGD Client and an SGD server, used for displaying applications.

To secure these connections, configure the SGD web server to be a secure (HTTPS) web server, and enable SGD security services. See [“Secure Connections to SGD Servers” on page 18](#) for details.

The SGD Secure Gateway can be used to provide an increased level of security between client devices and SGD servers. When you use the Gateway, client devices do connect directly to SGD. Instructions on how to install, configure, and use the SGD Gateway are included in the *Oracle Secure Global Desktop 4.6 Gateway Administration Guide*.

Connections Between SGD Servers and Application Servers

The connections between SGD servers and application servers are used to start applications on the application server, and to send and receive data from the application, such as key presses and display updates.

The level of security between SGD and your application servers depends on the types of application server and the protocols they use.

UNIX or Linux System Application Servers

When connecting using the Telnet protocol or the `rexec` command, all communication and passwords are transmitted unencrypted.

For secure connections to UNIX® or Linux system application servers, use Secure Shell (SSH). SSH encrypts all communications between SGD hosts and encrypts passwords before they are transmitted. See [“Using SSH” on page 178](#).

By default, SGD secures X displays using X authorization to prevent users from accessing X displays they are not authorized to access.

Microsoft Windows Application Servers

Windows applications use the Microsoft Remote Desktop (RDP) protocol. This means that all communication is encrypted, and connections to Microsoft Windows application servers are secure.

Web Application Servers

The level of security depends on the type of web server used to host the web application, as follows:

- **HTTP web servers** – All communication is unencrypted
- **HTTPS web server** – All communication is encrypted

For secure connections to web application servers, use HTTPS web servers.

Connections Between SGD Servers in an Array

Connections between SGD servers are used to share static and dynamic data across the array. See [“Replicating Data Across the Array” on page 334](#) for details of the information that is communicated on these connections. In a standard installation, the data transmitted between the SGD servers in an array is not encrypted. See [“Secure Intra-Array Communication” on page 336](#) for details on how to secure these connections.

DNS Names

The following are the main Domain Name System (DNS) requirements for SGD:

- Hosts must have DNS entries that can be resolved by all clients.
- DNS lookups and reverse lookups for a host must always succeed.
- All client devices must use DNS.

SGD servers can have multiple DNS names. Each SGD server has one peer DNS name, and one or more external DNS names.

Note – When configuring SGD, it is best to use fully-qualified DNS names.

A *peer DNS name* is the DNS name that the SGD servers in the array use to identify themselves to each other. For example, `boston.example.com`.

An *external DNS name* is the DNS name that the SGD Client uses to connect to an SGD server. For example, `www.example.com`.

These two types of DNS names might be associated with the same network interface on the SGD host, or they might each use a different network interface. These DNS names must be fully-qualified DNS names.

When you install SGD you are prompted for a DNS name for the SGD server. This must be the peer DNS name that is used inside the firewall. This is the DNS name that the SGD web server binds to.

After installation, you can configure each SGD server with one or more external DNS names. The external DNS name is used by the SGD Client when it connects to an SGD server. By default, the peer DNS name is also used as an external DNS name.

In a network containing a firewall, you might need to make some names usable outside the firewall, for example across the Internet, and others usable inside the firewall. For example, users outside the firewall might be able to use `www.example.com`, but not `boston.example.com`. Users inside the firewall might be able to use either name.



Caution – You do not have to make all your SGD servers available outside the firewall. However, if users log in to an SGD server from both inside and outside the firewall, they might not be able to resume some applications when logging in from outside the firewall.

If you use the SGD Gateway, client devices do connect directly to SGD, instead they connect using the DNS name of a Gateway or load balancer. External DNS names are only used for direct client connections that are not routed through the Gateway. Instructions on how to install, configure, and use the Gateway are included in the *Oracle Secure Global Desktop 4.6 Gateway Administration Guide*.

If you are using mechanisms such as an external hardware load balancer or round-robin DNS to control the SGD server that a user connects to, you must configure SGD to work with these mechanisms, see [“User Session Load Balancing” on page 353](#).

This section includes the following topics:

- [“Configuring External DNS Names” on page 5](#)
- [“Changing the Peer DNS Name of an SGD Server” on page 6](#)

Configuring External DNS Names

When an SGD Client connects directly to an SGD server, it connects using the external DNS name provided by the SGD server. The actual DNS name used is determined using the Internet Protocol (IP) address of the client.

If you use the SGD Gateway, external DNS names are only used for direct client connections that are not routed through an SGD Gateway.

You configure external DNS names by setting one or more filters that match client IP addresses to DNS names. Each filter has the format *Client-IP-Pattern:DNS-Name*

The *Client-IP-Pattern* can be either of the following:

- A regular expression matching one or more client device IP addresses, for example `192.168.10.*`
- A subnet mask expressed in the number of bits to match one or more client device IP addresses, for example `192.168.10.0/22`

SGD servers can be configured with several filters. The order of the filters is important because SGD uses the first matching *Client-IP-Pattern*.



Caution – If SGD is configured for firewall forwarding, you cannot use multiple external DNS names because SGD cannot determine the IP address of the client device. In this situation, you can configure a single external DNS name, for example `*:www.example.com`, and then use split DNS so that clients can resolve the name to different IP addresses, depending on whether they are inside or outside the firewall. See [“Firewall Traversal” on page 24](#).

The following is an example of external DNS names configuration:

```
"192.168.10.*:boston.example.com,*:www.example.com"
```

With this configuration, the following applies:

- Clients with IP addresses beginning `192.168.10` connect to `boston.example.com`.
- All other clients connect to `www.example.com`.

If the order of the filters is reversed, all clients connect to `www.example.com`.

▼ How to Configure the External DNS Names of an SGD Server

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

1. **In the Administration Console, go to the SGD Servers tab and select an SGD server.**

The General tab displays.

2. **In the External DNS Names field, type one or more filters for the external DNS names.**

Each filter matches client IP addresses to DNS names.

Press the Return key after each filter.

The format of each filter is described in [“Configuring External DNS Names” on page 5](#).

The order of the filters is important. The first match is used.

3. **Click Save.**

4. **Restart the SGD server.**

You must restart the SGD server for the external DNS names to take effect.

Changing the Peer DNS Name of an SGD Server

You can change the peer DNS name of an SGD server without having to reinstall the software, see [“How to Change the Peer DNS Name of an SGD Server” on page 7](#).

You must detach an SGD server from an array and stop SGD before changing its peer DNS name.

After changing the DNS name, the

`/opt/tarantella/var/log/SERVER_RENAME.log` file contains the details of the changes that were made. Your existing server security certificates are backed up in the `/opt/tarantella/var/tsp.OLD.number` directory.

If you use an SGD server as an application server, you must manually reconfigure the application server object by changing the DNS name for the application server and, optionally, renaming the object.

If you have installed SGD printer queues on UNIX or Linux platform application servers, you might have to remove the printer queue that uses the old DNS name of the SGD server, and configure a new printer queue that uses the new DNS name of the SGD server. See [“Configuring UNIX and Linux Platform Application Servers for Printing” on page 228](#).

▼ How to Change the Peer DNS Name of an SGD Server

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

You can only change the peer DNS name from the command line.

1. Log in as superuser (root) on the SGD host.

2. Detach the SGD server from the array.

If you are changing the peer DNS name of the primary SGD server, first make another server the primary server and then detach the server.

```
# tarantella array detach --secondary serv
```

Run the `tarantella status` command on the detached server to check that is detached from the array.

3. Stop the SGD server.

4. Ensure that the DNS name change for the SGD host has taken effect.

Check your DNS configuration and ensure that the other SGD servers can resolve the new DNS name. You might also have to edit the `/etc/hosts` and the `/etc/resolv.conf` files on the SGD host.

5. Change the DNS name of the SGD server.

Use the following command:

```
# tarantella serverrename --peerdns newname [ --extdns newname ]
```

It is best to use fully-qualified DNS names.

Use the `--extdns` option to change the external DNS name of the server. This option only works if the SGD server has a single external DNS name. If the server has more than one external DNS name, you must manually update the external DNS names. See [“Configuring External DNS Names” on page 5](#).

When prompted, type Y to proceed with the name change.

6. Regenerate the certificates used for secure intra-array communication.

```
# tarantella security keystoregen
```

For details about secure intra-array communication, see [“Secure Intra-Array Communication” on page 336](#).

If you are using the SGD Gateway, you must install the new peer Certificate Authority (CA) certificate on each SGD Gateway.

7. (Optional) Replace the server SSL certificate.

If the new peer DNS name is not included in the SSL certificate used by the SGD server you must replace the certificate, see [“How to Replace a Server SSL Certificate” on page 23](#).

If you are using the SGD Gateway, you must install the new server SSL certificate on each SGD Gateway.

8. Restart the SGD web server and SGD server.

9. Join the SGD server to the array.

The clock on the server joining the array must be in synchronization with the clocks on the other servers in the array. If the time difference is more than one minute, the array join operation fails.

```
# tarantella array join --primary p-serv --secondary s-serv
```

Proxy Servers

To be able to connect to SGD through a proxy server, client devices might need to be configured with the address and port number of the proxy servers. You might also need to configure SGD to give clients information about server-side proxy servers.

This section includes the following topics:

- [“Supported Proxy Servers” on page 8](#)
- [“Configuring Client Proxy Settings” on page 9](#)
- [“Proxy Server Timeouts” on page 11](#)
- [“Configuring Server-Side Proxy Servers” on page 11](#)

Supported Proxy Servers

The supported proxy servers are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

Configuring Client Proxy Settings

To configure client proxy settings, you must configure proxy settings for both the HTTP connections and the AIP connections. How you do this is described in the following sections.

HTTP Connections

HTTP connections are the connections between the user's browser and the SGD web server, for example to display a webtop. These connections always use the proxy settings configured for the browser.

AIP Connections

AIP connections are the connections between the SGD Client and the SGD server used to display applications. For these connections, the settings in the client profile control whether the SGD Client determines the proxy settings from a browser or from the client profile itself.

The SGD Client always stores the last proxy settings it used in the client profile cache. See [“About the Profile Cache” on page 317](#) for details.

Note – You can only configure a SOCKS proxy for the AIP connection by specifying an array route, see [“Configuring Server-Side Proxy Servers” on page 11](#) for details.

Determining Proxy Settings From a Browser

If the Use Default Web Browser Settings check box is selected in the client profile, the proxy server settings are determined from the user's default browser. The SGD Client stores the proxy settings in the profile cache on the client device and uses these settings when it next starts.

If Establish Proxy Settings on Session Start is selected in the client profile, the SGD Client obtains the proxy settings from the browser every time it starts. The stored proxy settings are not used. If Automatic Client Login is selected in the client profile, the Establish Proxy Settings on Session Start setting is disabled.

If the SGD Client is Integrated mode, and there are no proxy settings in the profile cache, the SGD Client attempts to make a direct connection.

To be able to determine the proxy settings from a browser, the browser must have Java technology enabled. If Java technology is not available, or it is disabled in the browser, the proxy settings must be manually specified in the client profile.

Note – If proxy server settings are defined in the Java Control Panel for the Sun Java Plugin tool, these settings are used instead of the browser settings.

Specifying Proxy Settings in the Client Profile

If the Manual Proxy Settings check box is selected in the client profile, you can specify either an HTTP or an SSL proxy server in the client profile itself.

Using Proxy Server Automatic Configuration Scripts

Whenever client proxy server configuration is determined from a browser, you can use an automatic configuration script to automatically configure the proxy settings.

You specify the Uniform Resource Locator (URL) of the configuration script in the connection settings for the browser. The automatic configuration script must be written in the JavaScript™ programming language and have either a .pac file extension or *no file extension*. See [Proxy Auto-Config File](#) for details.

Note – Use this format for all browsers supported by SGD.

Proxy Server Exception Lists

You can use proxy server exception lists to control the connections that are not proxied. Proxy exception lists can only be used if the proxy settings are determined from a browser. You cannot configure exception lists in the client profile. The exception list can be configured in the browser or Sun Java Plugin tool.

An exception list is a list of DNS host names. For Internet Explorer, the list is a semicolon-separated list. For Mozilla-based browsers, the list is a comma-separated list. Exception lists can include the * wildcard.

There is no translation between DNS host names and IP addresses in exception lists. For example, with an exception list of *.example.com, connections to chicago.example.com and detroit.example.com do not use a proxy server, but connections that use the IP addresses for these hosts do use a proxy server.

Exception lists must always include the following entries:

```
localhost; 127.0.0.1
```

Proxy Server Timeouts

Proxy servers can drop a connection after a short period of time if there is no activity on the connection. By default, SGD sends AIP keepalive packets every 100 seconds to keep the connection open.

If you find that applications disappear after a short while, you might have to increase the frequency at which AIP keepalive packets are sent.

In the Administration Console, go to the Global Settings → Communication tab and decrease the AIP Keepalive Frequency. Alternatively, use the following command:

```
$ tarantella config edit --sessions-aipkeepalive secs
```

Note – You must restart every SGD server in the array for changes to this attribute to take effect.

Configuring Server-Side Proxy Servers

SGD can be configured so that the SGD Client connects through a server-side SOCKS version 5 proxy server. The actual proxy server used is determined using the IP address of the client. This known as an *array route*.

If you use the SGD Gateway, array routes are only used for client connections that are not routed through an SGD Gateway.

You configure array routes by setting one or more filters that match client IP addresses to server-side proxy servers. Each filter has the format *Client-IP-Pattern:type:host:port*.

The *Client-IP-Pattern* can be either of the following:

- A regular expression matching one or more client IP addresses, for example `192.168.10.*`
- A subnet mask expressed in the number of bits to match one or more client IP addresses, for example `192.168.10.0/22`

The *type* is a connection type. Use CTSOCKS for a SOCKS version 5 connection. Use CTDIRECT to connect directly without using a proxy server.

The *host* and *port* are the DNS name or IP address and port of the proxy server to use for the connection.

SGD can be configured with several filters. The order of the filters is important because SGD uses the first matching *Client-IP-Pattern*.

If you use an external SSL accelerator instead of SGD to handle SSL processing, append the array route with `:ssl`, see the following example. This instructs the SGD Client to use SSL on that connection before continuing with the SOCKS connection. See [“Using External SSL Accelerators” on page 40](#) for details.



Caution – If SGD is configured for firewall forwarding, you cannot use multiple array routes because SGD cannot determine the IP address of the client device. You can configure a single array route, for example `*:CTSOCKS:taurus.example.com:8080`. See [“Firewall Traversal” on page 24](#).

The following is an example of array routes configuration:

```
192.168.5.*:CTDIRECT: \  
192.168.10.*:CTSOCKS:taurus.example.com:8080 \  
*:CTSOCKS:draco.example.com:8080:ssl
```

With this configuration, the following applies:

- Clients with IP addresses beginning `192.168.5` have a direct connection.
- Clients with IP addresses beginning `192.168.10` connect using the SOCKS proxy server `taurus.example.com` on port 8080.
- All other clients connect using the SOCKS proxy server `draco.example.com` on port 8080. These clients also connect using SSL before continuing with the SOCKS connection.

▼ How to Configure Array Routes

You can only configure array routes from the command line.

Ensure that no users are logged in to the SGD servers in the array, and that there are no running application sessions, including suspended application sessions.

1. Configure the filters for array routes.

Use the following command:

```
$ tarantella config edit \  
--tarantella-config-array-netservice-proxy-routes routes
```

Enclose *routes* in quotes and separate each filter with a comma, for example "filter1,filter2,filter3".

The format of each filter is described in [“Configuring Server-Side Proxy Servers” on page 11](#).

The order of the filters is important. The first match is used.

2. Restart every SGD server in the array.

You must restart every server in the array for array routes to take effect.

Firewalls

Firewalls can be used to protect various parts of a network and must be configured to allow the connections required by SGD.

This section includes the following topics:

- [“Firewalls Between Client Devices and SGD Servers” on page 13](#)
- [“Firewalls Between SGD Servers” on page 14](#)
- [“Firewalls Between SGD Servers and Application Servers” on page 15](#)
- [“Other Firewalls” on page 17](#)

Firewalls Between Client Devices and SGD Servers

Client devices must be able to make HTTP and AIP connections to any SGD server in the array. This is because a user’s SGD session and a user’s application sessions can be hosted on different SGD servers.

The following table lists the ports you might need to open to allow connections between client devices and SGD servers.

| Source | Destination | Port | Protocol | Purpose |
|--------|----------------|------|----------|---|
| Client | SGD web server | 80 | TCP | Standard, unencrypted HTTP requests and responses. Used to display webtops and for web services. |
| Client | SGD web server | 443 | TCP | Secure, encrypted HTTPS requests and responses. Used to display webtops and for web services. |
| Client | SGD server | 3144 | TCP | Standard, unencrypted AIP connections. Used for control and application display updates. |
| Client | SGD server | 5307 | TCP | SSL-based secure, encrypted AIP connections. Used for control and application display updates. |

Transmission Control Ports (TCP) 80 and 443 are the Internet-standard ports for HTTP and HTTPS. Port 443 is only used if HTTPS is enabled on the SGD web server. You can configure the SGD web server to use any port.

In a default installation, ports 3144 and 5307 must both be open in the firewall. The SGD Client initially makes a secure connection on port 5307, but once the user has authenticated, the connection is downgraded to a standard connection on port 3144. See [“Firewall Traversal” on page 24](#) for how to configure SGD when these ports cannot be opened.

If you enable SGD security services and use only HTTPS, only ports 443 and 5307 must be open in the firewall.

Ports 3144 and 5307 are registered with the Internet Assigned Numbers Authority (IANA) and are reserved for use only by SGD.

Firewalls Between SGD Servers

A network might contain firewalls between the SGD servers in an array, for example if you have multiple offices each containing an SGD server. The SGD servers in an array must be able to connect to any other member of the array.

The following table lists the ports you might need to open to allow connections between SGD Servers.

| Source | Destination | Port | Protocol | Purpose |
|------------|--------------------|----------------|----------|--|
| SGD server | Another SGD server | 515 | TCP | Used when moving print jobs from one SGD server to another using the <code>tarantella print move</code> command. |
| SGD server | Another SGD server | 1024 and above | TCP | Used to support audio, smart cards and serial ports for Windows applications. |
| SGD server | Another SGD server | 5427 | TCP | Used for connections between SGD servers to allow array replication, and sharing of both static and dynamic data across the array. |

Port 5427 is registered with IANA and is reserved for use only by SGD.

If you enable support for audio, smart cards, or serial ports for Windows applications, your firewall must allow connections between SGD servers on TCP port 1024 and above. The protocol engines that manage these features run on the SGD server that hosts the user session and this might be a different server to the one that hosts the application session. If you do not use these features, it is best to disable support for them in SGD. See the following for more information:

- [“How to Enable the SGD Windows Audio Service” on page 279](#)
- [“How to Enable Smart Cards in SGD” on page 294](#)
- [“How to Enable Access to Serial Ports” on page 299](#)

Firewalls Between SGD Servers and Application Servers

An SGD server must be able to connect to an application server in order to run applications.

The ports used for connections between SGD servers and application servers depends on the application type and the connection method used to log in to the application server. Other ports are needed to provide support while using applications.

The following table lists the ports you might need to open to allow connections between SGD Servers and application servers.

| Source | Destination | Port | Protocol | Purpose |
|--------------------|--------------------|----------------|----------|--|
| SGD server | Application server | 22 | TCP | Used to connect to X and character applications using SSH. |
| SGD server | Application server | 23 | TCP | Used to connect to Windows, X, and character applications using Telnet. |
| Application server | SGD server | 139 | TCP | Used for UNIX and Linux platform client drive mapping services. The server binds to this port at start-up, whether or not client drive mapping services are enabled. |
| SGD server | Application server | 512 | TCP | Used to connect to X applications using rexec. |
| Application server | SGD server | 515 | TCP | Used to send print jobs from the application server to an SGD server. |
| SGD server | Application server | 3389 | TCP | Used to connect to Windows applications that use the Microsoft RDP protocol. |
| SGD server | Application server | 3579 | TCP | Used for connections between the primary SGD server and the SGD load balancing service on an application server. |
| Application server | SGD server | 3579 | UDP | Used for connections between the SGD load balancing service on an application server and the primary SGD server. |
| SGD server | Application server | 5999 | TCP | Used to connect to Windows applications, if the application is configured to use the Wincenter protocol <i>and</i> the connection method is Telnet. The Wincenter protocol is no longer supported but might be used by legacy Windows application objects. |
| Application server | SGD server | 6010 and above | TCP | Used to connect X applications to the protocol engines on the SGD server. |

For X applications, ports 6010 and above are only used if the connection method for X applications is Telnet or rexec. If the connection method is SSH, the connections use port 22. If you enable audio for X applications, all ports must be open between the application server and SGD. This is because the SGD audio daemon connects to the SGD server on random ports. This applies even if the connection method is SSH. See [“Audio” on page 276](#) for details.

Port 3579 is registered with IANA and is reserved for use only by SGD. You only need to open these ports if you are using SGD Advanced Load Management. See [“Application Load Balancing” on page 362](#) for details.

Other Firewalls

SGD needs to make connections to any authentication services and directory services you might be using.

The following table lists the ports you might need to open to allow connections between SGD Servers and other services.

| Source | Destination | Port | Protocol | Purpose |
|--------------------------------|--------------------------------|---------------|------------|---|
| SGD server | Windows server | 88 | TCP or UDP | Used to authenticate users in an Active Directory forest. |
| SGD server | Windows server | 137 | UDP | Used to authenticate users in a Microsoft Windows domain. |
| SGD server | Windows server | 139 | TCP | Used to authenticate users in a Microsoft Windows domain. |
| SGD server | LDAP directory server | 389 | TCP | Used to authenticate users, or to assign applications to users, using a Lightweight Directory Access Protocol (LDAP) directory. |
| SGD server | Windows server | 464 | TCP or UDP | Used to enable users to change their password if it has expired. |
| SGD server | LDAP directory server | 636 | TCP | Used to authenticate users, or to assign applications to users, using a secure connection (LDAPS) to an LDAP directory. |
| SecurID Authentication Manager | SGD server | 1024 to 65535 | UDP | Used to authenticate users using SecurID. |
| SGD server | Windows server | 3268 | TCP | Used to authenticate users in an Active Directory forest. |
| SGD server | Windows server | 3269 | TCP | Used to authenticate users in an Active Directory forest. |
| SGD server | SecurID Authentication Manager | 5500 | UDP | Used to authenticate users using SecurID. |

Ports 88, 464, 3268, 3269 are only required if you are using Active Directory authentication. Ports 88 and 464 can use either the TCP or UDP protocol depending on the packet size and your Kerberos configuration. See [“Configuring SGD for Kerberos Authentication” on page 56](#) for details. Ports 3268 and 3269 are only used for SSL connections to Active Directory, see [“SSL Connections to Active Directory” on page 56](#) for details.

Ports 137 and 139 are only required if you are using a domain controller for authentication. See [“Windows Domain Authentication” on page 83](#) for details.

Ports 389 and 636 are only required if you are using an LDAP directory to establish a user's identity or to assign applications to users. This applies to the following authentication mechanisms:

- Active Directory authentication, see [“Active Directory Authentication” on page 52](#)
- LDAP authentication, see [“LDAP Authentication” on page 63](#)
- Third-party using the LDAP search method, see [“Third-Party Authentication” on page 72](#)

Ports 1024 to 65535 are only required if you are using SecurID Authentication. For the RSA SecurID Authentication Manager to communicate with an SGD server acting as an Agent Host, all ports from 1024 to 65535 must be open from the IP addresses of the Master and Slave Authentication Managers to the IP addresses of all Agent Hosts. See [“SecurID Authentication” on page 69](#) for details.

Port 5500 is only required if you are using SecurID authentication. For the RSA SecurID Authentication Manager to communicate with an SGD server acting as an Agent Host, port 5500 must be open from the IP addresses of the Host Agents to the IP addresses of the Master and Slave Authentication Managers.

Secure Connections to SGD Servers

When SGD is first installed, the HTTP connections to the SGD web server are not secure. The initial AIP connection to an SGD server is secure, but after the user is logged in, the AIP connection is downgraded to a standard connection.

You secure AIP connections by installing an SSL certificate on the SGD server and enabling an SGD security services. SGD security services enables SGD to use Transport Layer Security (TLS) or Secure Sockets Layer (SSL) to provide secure connections to an SGD server.

To secure HTTP connections, HTTPS connections must be enabled on the SGD web server. You enable HTTPS connections by using the `--https` argument when you start the SGD server or the SGD web server. The SGD web server is preconfigured to be a HTTPS web server and use the same SSL certificate as the SGD server. If you enable HTTPS connections, you must enable HTTPS connections for every SGD web server in the array. You must not mix HTTP and HTTPS web servers in the same SGD array and every SGD web server in the array must use the same HTTPS port.

Once you enable secure connections, ensure that users have an HTTPS URL for the Login URL in their client profile. See [“Client Profiles” on page 311](#).

You can enable secure connections using automatic configuration or manual configuration. Automatic configuration is the easiest to use.

This section includes the following topics:

- [“SSL Certificates” on page 19](#)
- [“Firewall Traversal” on page 24](#)
- [“Enabling Secure Connections \(Automatic Configuration\)” on page 25](#)
- [“Enabling Secure Connections \(Manual Configuration\)” on page 28](#)
- [“Secure Connections and Security Warnings” on page 33](#)

SSL Certificates

An SSL certificate is an encoded file that a secure service, such as a web server, uses to identify itself to a client. When secure connections are enabled, an SGD server requires an SSL certificate.

The SGD web server is preconfigured to use the same SSL certificate as the SGD server. This is configured in the Apache configuration file, `/opt/tarantella/webserver/apache/apache-version/conf/httpd.conf`. You can use a separate SSL certificate for the SGD web server if you prefer.

SGD supports Privacy Enhanced Mail (PEM) Base 64-encoded X.509 certificates. These certificates have the following structure:

```
-----BEGIN CERTIFICATE-----  
...certificate...  
-----END CERTIFICATE-----
```

SGD supports the Subject Alternative Name (`subjectAltName`) extension for server SSL certificates. This enables you to associate more than one DNS name with a certificate. If the `subjectAltName` field is present in an SSL certificate, the `subject` field is ignored and only the `subjectAltName` is used. The SSL certificate is accepted by the SGD Client if any of the Subject Alternative Names match the name of the SGD server you are connecting to.

Supported Certificate Authorities

A server SSL certificate is issued by a Certificate Authority (CA). A CA is a trusted third party that digitally signs a server SSL certificate using a CA, or root, certificate.

SGD includes support for a number of CA certificates by default. The `/opt/tarantella/etc/data/cacerts.txt` file contains the X.500 Distinguished Names (DNs) and MD5 signatures of all the CA certificates that SGD supports.

If you need to create a certificate signing request (CSR) for signing by a CA, see [“How to Generate a Certificate Signing Request” on page 20](#).

You can use a server SSL certificate that is signed by an unsupported CA. However, by default, all users are prompted to accept or decline these certificates because they cannot be validated by SGD. This is a potential security risk. See [“Secure Connections and Security Warnings” on page 33](#) for more details.

SGD supports the use of certificate chains. With certificate chains, an Intermediate CA signs an SSL certificate with a CA certificate that is issued by another CA.

If your server SSL certificate is signed by an unsupported CA, or an Intermediate CA, you must install the CA certificate or certificate chain.

Self-Signed SSL Certificates

SGD enables you to create self-signed server SSL certificates for test purposes, for example, while you are waiting to complete the registration requirements before the SSL certificate can be generated. Self-signed certificates are valid for 365 days.

Only use self-signed SSL certificates in a test environment because self-signed SSL certificates are not truly secure. While a self-signed SSL certificate can be used to give users secure connections, users have no guarantee that the server they are connecting to is genuine.

You can create self-signed SSL certificates with the following commands:

- `tarantella security selfsign` – enables you to self-sign a CSR generated with the `tarantella security certrequest` command
- `tarantella security enable` – enables you to configure a secure SGD server automatically and install a server SSL certificate

Using an SSL Certificate Obtained for Another Product

You can use an SSL certificate that was originally obtained for another product, such as a web server. To do this, you must have the private key for that certificate. If the private key is encrypted by a product that uses the SSLeay or OpenSSL certificate libraries, you can obtain the private key by decrypting it with the `tarantella security decryptkey` command. If you do not have the private key, you must obtain a new server SSL certificate.

▼ How to Generate a Certificate Signing Request

1. **Log in as superuser (root) on the SGD host.**

2. Generate a CSR.

Use the `tarantella security certrequest` command to generate the CSR.

SGD supports the Subject Alternative Name (`subjectAltName`) extension for server SSL certificates. This enables you to associate more than one DNS name with an SSL certificate. See [“DNS Names” on page 3](#) for details.

SGD supports the use of the `*` wildcard for the first part of the domain name, for example `*.example.com`.

Generating the CSR also creates the public and private key pair.

On the SGD server, the CSR is stored in the `/opt/tarantella/var/tsp/csr.pem` file, and the private key is stored in the `/opt/tarantella/var/tsp/key.pending.pem` file.

If you are replacing a server SSL certificate, for example because it is about to expire, you can generate a CSR without affecting your current certificate.

In the following example, a CSR is generated for the SGD server `boston.example.com`. This server also has an external DNS name of `www.example.com` and so this name is added as a Subject Alternative Name.

```
# tarantella security certrequest \  
--country US --state Massachusetts --orgname "Example Com"
```

The certificate's common name (CN) will be: `boston.example.com`

This hostname is included in the Certificate Signing Request (CSR) and corresponds to the name of the server that users will connect to.

- If DNS names are used to connect to the server, the hostname above MUST be a fully qualified DNS name.
- If clients are required to connect to the server using an IP address, the hostname above should be the IP address. A DNS record for this IP address SHOULD NOT exist.

For clients to accept the certificate once it's installed, a DNS lookup of the hostname followed by a reverse lookup of the result must return the original hostname.

The hostname to be used in the certificate request is `boston.example.com`.

Do you want to use this hostname? [yes] **y**

Do you want to add any additional hostnames? [no] **y**

Type in the subject alternative names for the certificate, one per line. Enter a blank line to finish.

subjectAltName: **www.example.com**

subjectAltName:

2048 semi-random bytes loaded

Generating RSA private key, 1024 bit long modulus

..+++++

.....+++++

e is 65537 (0x10001)

Certificate Signing Request (CSR): Summary

Subject:

C=US

ST=Massachusetts

O=Example Com

CN=boston.example.com

Subject Alternative Names:

DNS:boston.example.com, DNS:www.example.com

The information above will be contained in the CSR.

Create CSR now? [yes] **y**

Send the following Certificate Signing Request (CSR) to a Certificate Authority, such as VeriSign (www.verisign.com). Check with your CA that you're providing all the information they need.

-----CUT HERE-----

-----BEGIN CERTIFICATE REQUEST-----

NhY2h1c2V0MIIB5TCCAUAQAwXDELMAkGA1UEBhMCVVMxGjAUBgNVBAGTDU1hc3dpZS5VdHMxGTAXBgNVBAoTEEluZGlnbyBJbnN1cmFuY2UxGjAYBgNVBAMTEXJhZGsd+SmX7zz6Sy5TdW4uQ09NMIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDbWMsMqBs7gQw8Q1Gk3NAypySP6aRiEitLrfs1Z8XgKXmjmlLtb03V9JonjLfH3fBzkgN0G6EpTmJM4y30OpEXZZ2yhtWgsQsXyLWbfTLWZPfHLPI5ztEEJ7Z0G6dpeG0xgwODA2ApAp6sIrmBqbZG2Aaf5poB+FQ41smQIDAQABoEkWRwYJKoZIhvcNAQkOMToN1cmFuBgNVHREELzAtghFyYWRnaWUuVU5uU3VuLkNPTyIYd3d3LmluZGlnbyl1bnV617E7oFKY2UuY29tMA0GCSqGSIb3DQEBAQUAA4GBAMsOieZzrGHN7fkW6LmYNHWSW1tmHeFjekpiUiTLYE+KUZXXXCH9fleo+nfwFdi9VOomIdga4uehl+4acqqiGEeW4iIb9BC9b/V1pA/lGJwWN0aDDB3/d47UGAlli+spW37chg53Fp7eP2xIYWfJR6035eSPZm42dyp

-----END CERTIFICATE REQUEST-----

-----CUT HERE-----

When you receive your certificate, use 'tarantella security certuse' to install it.

3. Send the CSR to a CA.

See [“Supported Certificate Authorities” on page 19](#) for details of the CAs that SGD supports by default.

Either copy the CSR as output from the command line, or use the copy of the CSR that is stored in the `/opt/tarantella/var/tsp/csr.pem` file on the SGD server.

▼ How to Replace a Server SSL Certificate

Use the following procedure to replace the server SSL certificate for an SGD server, for example because the original SSL certificate is about to expire.

1. (Optional) Generate a CSR and send it to a CA.

See [“How to Generate a Certificate Signing Request” on page 20](#).

2. Install the server SSL certificate.

See [“How to Install a Server SSL Certificate” on page 29](#).

3. (Optional) Install the CA certificate.

Perform this step only if the server SSL certificate is signed by an unsupported CA, or an Intermediate CA, see [“Supported Certificate Authorities” on page 19](#).

The certificates that must be installed are as follows:

- **Unsupported CA.** Import the CA or root certificate, see [“How to Install the CA Certificate for an Unsupported CA” on page 30](#).
- **Intermediate CA.** Import the CA certificate chain, see [“How to Install a CA Certificate Chain” on page 30](#).

4. Restart the SGD server and SGD web server.

You must restart the SGD server to ensure that the new server SSL certificate is used for secure connections.

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

Use the following command:

```
# tarantella restart
```

Firewall Traversal

AIP connections to an SGD server are made on TCP ports 3144 and 5307. If it is not possible to open the required ports in your firewalls, you can direct all SGD traffic through a single port, usually port 443. To do this, you can use either of the following:

- **The SGD Gateway** – See [“The SGD Gateway” on page 24](#) for details
- **Firewall forwarding** – See [“Using Firewall Forwarding” on page 24](#) for details

The SGD Gateway is the best solution for traversing firewalls, and has other benefits such as load-balanced HTTP connections.

The SGD Gateway

The SGD Gateway is an optional SGD component. It is a proxy server designed to be deployed in front of an SGD array in a demilitarized zone (DMZ), and it enables an SGD array to be located on your internal network. Additionally, all connections can be authenticated in the DMZ before any connections are made to the SGD servers in the array.

Instructions on how to install, configure, and use the SGD Gateway are included in the *Oracle Secure Global Desktop 4.6 Gateway Administration Guide*.

Using Firewall Forwarding

If you are not using the SGD Gateway, you can use *firewall forwarding* to give users access to SGD using a single port. With firewall forwarding, you configure the SGD server to listen on port 443. The SGD server then forwards all traffic that is not AIP traffic to the SGD web server.

If SGD is configured for firewall forwarding, you cannot use any SGD features that depend on filtering the IP address of the client device. This means you cannot use the following features:

- **The SGD Gateway** – See [“The SGD Gateway” on page 24](#) for details
- **Multiple external DNS names** – See [“Configuring External DNS Names” on page 5](#) for details
- **Multiple array routes** – See [“Configuring Server-Side Proxy Servers” on page 11](#) for details
- **Connection definitions** – See [“Using Connection Definitions” on page 43](#) for details

If you use firewall forwarding, you can configure a single external DNS name, for example `*:www.example.com`, and then use split DNS so that clients can resolve the name to different IP addresses, depending on whether they are inside or outside the firewall.

Enabling Secure Connections (Automatic Configuration)

The `tarantella security enable` command enables you to quickly configure and enable secure connections. You can only use this command if *both* of the following are true:

- The SGD installation is a fresh installation and no attempt has been made to configure secure connections.
- The SGD server is not joined with other SGD servers in an array.

If these conditions are not met, the `tarantella security enable` command fails and you must enable security by configuring it manually. See [“Enabling Secure Connections \(Manual Configuration\)” on page 28](#).

The `tarantella security enable` command performs the following configuration:

- Installs the specified server SSL certificate.
- Enables HTTPS connections to the SGD web server.
- (Optional) Configures the SGD server for firewall forwarding.
- Enables SGD security services.
- Restarts the SGD server and SGD web server.

To install an SSL certificate, you must already have the certificate and private key. If you need to submit a CSR for signing by a CA, see [“How to Generate a Certificate Signing Request” on page 20](#).

If you do not specify a server SSL certificate to install, the `tarantella security enable` command creates and installs a self-signed SSL certificate. If you want to install a server SSL certificate later, use the `tarantella security disable` command to restore the security settings to their previous state. You can then run the `tarantella security enable` command again and specify a server SSL certificate.

By default, the `tarantella security enable` command configures the SGD server for firewall forwarding. Use the `--firewalltraversal off` option if you want to enable security without firewall forwarding. You can enable firewall forwarding later by doing either of the following:

- Use the `tarantella security disable` command to restore the security settings to their previous state. Then use `tarantella security enable` to configure the SGD server for firewall forwarding.
- Enable firewall forwarding manually. See [“How to Configure Firewall Forwarding” on page 31](#) for details of how to do this.



Caution – SGD servers configured for firewall forwarding cannot be used with the SGD Gateway.

▼ How to Enable Secure Connections (Automatic Configuration)

Before you begin, ensure you have access to the server SSL certificate, and the private key and CA certificate, if needed. The certificates must be in PEM format.

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

Ensure that the SGD server is running. You can use the `tarantella status` command to show the current status of an SGD server.

1. Log in as superuser (root) on the SGD host.

2. Install a server SSL certificate and enable SGD security services.

Use the `tarantella security enable` command to install a server SSL certificate and enable SGD security services.

If you used the `tarantella security certrequest` command to generate a CSR, you can omit the `--keyfile` option. The key stored in the `/opt/tarantella/var/tsp/key.pending.pem` file is used. When you install the server SSL certificate, the private key is moved to the `/opt/tarantella/var/tsp/key.pem` file.



Caution – If you use the `--certfile` option and the `--keyfile` option together, SGD creates *symbolic links* to the SSL certificate file and the key file in the `/opt/tarantella/var/tsp` directory on the SGD server. Do not delete or move the SSL certificate file or key file after running this command.

If you do not specify a server SSL certificate to install, the `tarantella security enable` command generates a CSR, and then creates and installs a self-signed SSL certificate. Only use a self-signed SSL certificate for test purposes.

SGD supports a number of CAs by default. Only use the `--rootfile` option if the server SSL certificate is signed by an unsupported CA, or by an Intermediate CA. See “Supported Certificate Authorities” on page 19 for details.

If the server SSL certificate is signed by an Intermediate CA, combine all the certificates in the CA certificate chain into a file. The certificates must be in PEM format. The CA certificate used to sign the server SSL certificate *must appear first*, for example:

```
-----BEGIN CERTIFICATE-----
...Intermediate CA's certificate...
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
...CA root certificate...
-----END CERTIFICATE-----
```

If you specify the path to a certificate or key file, you must specify the *full path* to the file.



Caution – If you use the SGD Gateway, you must use the `--firewalltraversal off` option to enable security without firewall forwarding. SGD servers configured for firewall forwarding cannot be used with the SGD Gateway.

The CSR, the SSL certificate, the private key, and the CA certificate are stored in the `/opt/tarantella/var/tsp` directory on the SGD server.

- If the server SSL certificate is signed by a supported CA, and the `tarantella security certrequest` command was used to generate a CSR, use the following command:

```
# tarantella security enable [ --firewalltraversal off ] \  
--certfile certificate-path
```

- If the server SSL certificate is signed by a supported CA, and the `tarantella security certrequest` command was *not* used to generate a CSR, use the following command:

```
# tarantella security enable [ --firewalltraversal off ] \  
--certfile certificate-path --keyfile key-path
```

- If the server SSL certificate is signed by an unsupported CA, or an Intermediate CA, use the following command:

```
# tarantella security enable [ --firewalltraversal off ] \  
--certfile certificate-path [--keyfile key-path] \  
--rootfile CA-certificate-path
```

- To enable SGD security services with a self-signed SSL certificate, use the following command:

```
# tarantella security enable [ --firewalltraversal off ]
```

Enabling Secure Connections (Manual Configuration)

Enabling secure connections with manual configuration involves the following steps:

1. (Optional) Generate a Certificate Signing Request (CSR) and send it to a CA.

See [“How to Generate a Certificate Signing Request” on page 20](#).

This step is optional if you obtain a server SSL certificate *without* using the `tarantella security certrequest` command to generate a CSR.

If you already have an SSL certificate for another product, such as a web server, you might be able to use that certificate. See [“Using an SSL Certificate Obtained for Another Product” on page 20](#).

2. Install an SSL certificate for each SGD server in the array.

To use secure connections, an SGD server must present an SSL certificate to identify itself to an SGD Client. See [“How to Configure the External DNS Names of an SGD Server” on page 6](#).

3. (Optional) Install the CA certificate.

Perform this step only if the server SSL certificate is signed by an unsupported CA, or an Intermediate CA, see [“Supported Certificate Authorities” on page 19](#).

The certificates that must be installed are as follows:

- **Unsupported CA.** Import the CA or root certificate, see [“How to Install the CA Certificate for an Unsupported CA” on page 30.](#)
 - **Intermediate CA.** Import the CA certificate chain, see [“How to Install a CA Certificate Chain” on page 30.](#)
4. (Optional) Configure SGD for firewall forwarding.

For details of when to use firewall forwarding, see [“Firewall Traversal” on page 24.](#)

See [“How to Configure Firewall Forwarding” on page 31.](#)
 5. Enable SGD security services and restart SGD.

To enable secure connections, you must enable SGD security services and restart SGD.

See [“How to Enable SGD Security Services for an SGD Server” on page 32.](#)

▼ How to Install a Server SSL Certificate

Use the following procedure to install a server SSL certificate that was obtained by using the `tarantella security certrequest` command to generate a CSR.

Before you begin, ensure you have access to the server SSL certificate. The SSL certificate must be in PEM format.

1. Log in as superuser (root) on the SGD host.

2. Install the server SSL certificate.

Use the `tarantella security certuse` command to install the SSL certificate.

If you are replacing a server SSL certificate, for example because the original SSL certificate is about to expire, the `tarantella security certuse` command prompts you for confirmation before overwriting the SSL certificate and private key.

When you install the server SSL certificate, the private key stored in the `/opt/tarantella/var/tsp/key.pending.pem` file is moved to the `/opt/tarantella/var/tsp/key.pem` file.

If you specify the path to a file, you must specify the *full path* to the file.

The CSR, the SSL certificate, and the private key are stored in the `/opt/tarantella/var/tsp` directory on the SGD server.

- To install the SSL certificate from standard input, use the following command:

```
# tarantella security certuse
```

Paste the server SSL certificate in to standard input and press Control-D.

- To install the SSL certificate from a temporary file, use the following command:

```
# tarantella security certuse < /tmp/cert
```

- To install the SSL certificate from a permanent file, use the following command:

```
# tarantella security certuse --certfile /opt/certs/cert.pem
```



Caution – This command creates a *symbolic link* to the SSL certificate file stored in the `/opt/tarantella/var/tsp` directory on the SGD server. Do not delete or move the SSL certificate file after running this command.

▼ How to Install the CA Certificate for an Unsupported CA

Before you begin, ensure you have access to the CA certificate. The CA certificate must be in PEM format.

1. Log in as superuser (root) on the SGD host.
2. Install the CA certificate.

Use the `tarantella security customca` command.

- To install the CA certificate from standard input, use the following command:

```
# tarantella security customca
```

Paste the CA certificate in to standard input and press Control-D.

- To install the CA certificate from a file, use the following command:

```
# tarantella security customca --rootfile /tmp/cert
```

▼ How to Install a CA Certificate Chain

Before you begin, ensure that you have all the certificates in the CA certificate chain. The certificates must be in PEM format.

1. Log in as superuser (root) on the SGD host.

2. Combine all the certificates in the chain into a file.

For example, create a file called `chainedcerts.pem`.

The CA certificate used to sign the server SSL certificate *must appear first*, for example:

```
-----BEGIN CERTIFICATE-----
...Intermediate CA's certificate...
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
...CA root certificate...
-----END CERTIFICATE-----
```

3. Install the CA certificate chain.

Use the `tarantella security customca` command.

- To install the CA certificate from standard input, use the following command:

```
# tarantella security customca
```

Paste the CA certificate chain in to standard input and press Control-D.

- To install the CA certificate from a file, use the following command:

```
# tarantella security customca --rootfile /tmp/chainedcerts.pem
```

▼ How to Configure Firewall Forwarding

1. Configure each SGD web server in the array to bind to localhost and TCP port 443.

Repeat the following step on each SGD web server in the array.

a. Log in as superuser (root) on the SGD host.

b. Edit the Apache configuration file.

The configuration file is `/opt/tarantella/webserver/apache/apache-version/conf/httpd.conf`.

c. Change the `<IfDefine SSL>` directive in the SSL Support section.

Change the directive to the following:

```
<IfDefine SSL>
Listen 127.0.0.1:443
</IfDefine>
```

d. Save the changes.

2. Log in as superuser (root) on the primary SGD server in the array.
3. Configure each SGD server in the array to use TCP port 443 for encrypted connections.

Use the following command:

```
# tarantella config edit --array-port-encrypted 443
```

Tip – You can also configure the port in the Administration Console. Go to the Global Settings → Communication tab. Type 443 in the Encrypted Connections Port field.

4. Configure each SGD server in the array to forward HTTP traffic to the SGD web server.

Use the following command:

```
# tarantella config edit --array \  
--security-firewallurl https://127.0.0.1:443
```

Tip – You can also configure the port in the Administration Console. Select an SGD server and go to the Security tab. Type https://127.0.0.1:443 in the Firewall Forwarding URL field.

5. Check that the firewall forwarding URL has taken effect for each SGD server in the array.

Use the following command to check each server:

```
# tarantella config list --server serv --security-firewallurl
```

▼ How to Enable SGD Security Services for an SGD Server

Ensure that the SGD server is running. You can use the `tarantella status` command to show the current status of an SGD server.

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

1. Log in as superuser (root) on the SGD host.

2. Enable SGD security services.

Use the following command:

```
# tarantella security start
```

3. Restart the SGD server and SGD web server.

Use the following command:

```
# tarantella restart --https
```

Once security is enabled, security services are available whenever SGD restarts.

Secure Connections and Security Warnings

When using secure connections to SGD, users see some or all of the following security warnings:

- Browser and Java Plugin tool security warnings
- SGD server SSL certificate security warnings
- Untrusted initial connection warnings

Note – Users might see these warnings *even if* SGD security services are not enabled. This is because the initial connection between an SGD Client and an SGD server is always secure.

This section describes why these warnings occur and what you can do about them.

Browser and Java Plugin Tool Security Warnings

If you have enabled secure connections (HTTPS) to the SGD web server, users see a warning if the CA or root certificate used to sign the web server SSL certificate is not available in the browser's certificate store. To enable the web server SSL certificate to be validated without displaying a security warning, import the CA or root certificate into the user's browser certificate store. Use the browser's tools to do this.

If Java technology is enabled in the browser, the Java Plugin tool might also warn users about the web server's SSL certificate. This depends on the configuration in the Java Control Panel. By default, the Java Plugin tool is configured to use the certificates in the browser certificate store. If the Plugin tool is not configured to do this, you might have to import the CA or root certificate using the Java Control Panel.

SGD Server SSL Certificate Security Warnings

When a user logs in to an SGD server that has a server SSL certificate, the SGD Client validates the certificate before proceeding.

If there is a problem with a server SSL certificate, users see a security warning message. The security warning message enables users to view the SSL certificate details before deciding to accept the SSL certificate permanently or temporarily, or to reject it. [FIGURE 1-1](#) shows an example security warning message.

FIGURE 1-1 Example SGD Server SSL Certificate Security Warning Message



If users reject the SSL certificate, the connection to SGD is terminated.

If users accept the SSL certificate temporarily, and they agree to the initial connection, the SSL certificate details are cached for the lifetime of the user session. When users next log in, they are prompted about the SSL certificate again. If users accept the SSL certificate permanently, and they agree to the initial connection, they are not prompted about the SSL certificate again. For details about agreeing to the initial connection, see [“Untrusted Initial Connection Warnings” on page 35](#).

Users see security warnings about SSL certificates in the following circumstances:

- **Invalid dates** – The current date is earlier than the Not Before date in the SSL certificate, or the current date is later than the Not After date in the SSL certificate
- **Incorrect host name** – The name of the host the SGD Client is connecting to does not match the Subject or Subject Alt Name in the SSL certificate
- **Issuer unknown** – The SSL certificate is signed by an unsupported CA

For details about how to avoid issuer unknown security warnings, see [“Avoiding Issuer Unknown Security Warnings” on page 37](#).

Untrusted Initial Connection Warnings

SGD requires users to authorize their connections to SGD so that they only connect to servers they trust. The first time a user connects to an SGD server, they see an Untrusted Initial Connection message advising that they are connecting to a server for the first time, as shown in [FIGURE 1-2](#).

FIGURE 1-2 Untrusted Initial Connection Warning



Users can check the SSL certificate details by clicking the View Certificate button and checking that the Validity and Subject details are correct. Users must do this *before* clicking Yes to agree to the connection. When a user agrees to a connection, the following files are updated on the client device:

- `hostsvisited`
- `certstore.pem`

The `hostsvisited` and `certstore.pem` files are stored in the same location as the user's client profile cache. See [“About the Profile Cache” on page 317](#) for details.

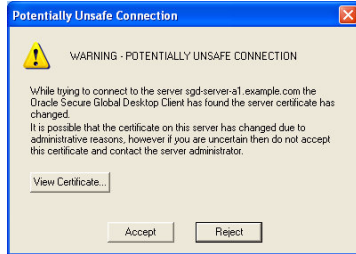
When a user agrees to a connection to an SGD server, the `hostsvisited` file on the client device is updated with the name of the SGD server. If the server SSL certificate is signed by an unsupported CA, the fingerprint of the CA certificate is also added. The user is not prompted about the connection again, unless there is a problem.

When a user agrees to a connection to an SGD server, and the server SSL certificate is valid, the server SSL certificate is added to the `certstore.pem` file on the client device.

If there is a problem with the server SSL certificate, for example because it is signed by an unsupported CA, users see a certificate security warning, as described in [“SGD Server SSL Certificate Security Warnings” on page 34](#). If a user permanently accepts the SSL certificate, or the SSL certificate and its CA chain, and agrees to the connection to an SGD server, the SSL certificate is added to the `certstore.pem` file on the client device. When the user next logs in, they *are not prompted* about the SSL certificate. If a user accepts the SSL certificate temporarily, and they agree to the connection to an SGD server, the SSL certificate is not added to the `certstore.pem` file on the client device. When the user next logs in, they *are prompted* about the SSL certificate.

If there is a problem with the connection, for example because the server SSL certificate has changed, a Potentially Unsafe Connection message displays, as shown in [FIGURE 1-3](#).

FIGURE 1-3 Potentially Unsafe Connection Message



To ensure that users only connect to SGD servers that are trusted, SGD Administrators can do the following:

- Explain to users the security implications of agreeing to a connection to an SGD server.
- Provide users with a preconfigured `hostsvisited` file. See [“Using a Preconfigured `hostsvisited` File” on page 36](#).

See also [“Avoiding Issuer Unknown Security Warnings” on page 37](#) for details of how to prevent users from seeing issuer unknown security warnings.

Using a Preconfigured `hostsvisited` File

A preconfigured `hostsvisited` file can be used to prevent users from seeing a warning when the SGD Client first connects to an SGD server. You can also use it to restrict the SGD servers to which the SGD Client can connect.

To use a preconfigured `hostsvisited` file, first create a file containing the host names of all the SGD servers. If the server SSL certificate for an SGD server is signed by an unsupported CA, you must also add the fingerprint of the CA certificate. The easiest way to do this is to copy and edit an existing `hostsvisited` file, and then install it on client devices. You can also obtain the CA certificate fingerprint using the `tarantella security fingerprint` command.

You can manually add an `<allowhostoverride>` line to the `hostsvisited` file. If the value of `<allowhostoverride>` line is 0, the SGD Client can only connect to SGD servers that have entries in the `hostsvisited` file. If the value of `<allowhostoverride>` line is 1, or if the `<allowhostoverride>` line is missing, the SGD Client can connect to any SGD server. Users only see a warning when the SGD Client connects to an SGD server that is not listed in the `hostsvisited` file.

The following is an example hostsvisited file.

```
<?xml version="1.0" encoding="UTF-8" ?>
<array>
  <allowhostoverride>0</allowhostoverride>
  <server peername="boston.example.com">
    <certfingerprint>51:B7:6D:FA:6E:3B:BE:ED:37:73:D4:9D:5B:C5:71:F6
  </certfingerprint>
  </server>
</array>
```

Avoiding Issuer Unknown Security Warnings

Issuer unknown security warnings occur when the server SSL certificate for an SGD server is issued by an unsupported CA. The warning displays because the issuer of certificate cannot be validated.

The easiest way to avoid issuer unknown security warnings is to ensure that a server SSL certificate is signed by a supported CA. See [“Supported Certificate Authorities” on page 19](#) for details.

To enable the SSL certificate to be validated, you must install the CA certificate or certificate chain. However, even if you install the CA certificate, users see a security warning about the SLL certificate the first time they connect to the SGD server. The only way to prevent users from being warned about the certificate is add the server SSL certificate to the `certstore.pem` file on the client device. The server SSL certificate is stored in the `/opt/tarantella/var/tsp/cert.pem` file on each SGD server.

Tuning Secure Connections to SGD Servers

This section describes the tuning that can be applied to secure connections to SGD servers and includes the following topics:

- [“Tuning the SSL Daemon” on page 38](#)
- [“Using External SSL Accelerators” on page 40](#)
- [“Selecting a Cipher Suite for Secure Connections” on page 41](#)
- [“Using Connection Definitions” on page 43](#)

Tuning the SSL Daemon

The SSL Daemon is the SGD component that handles secure connections to SGD servers. On the SGD host, the SSL Daemon is listed as one or more `ttssl` processes.

By default, the SSL Daemon listens on TCP port 5307 for AIP traffic that has been encrypted with SSL. However, if you are using firewall forwarding, the SSL Daemon listens on port 443, and accepts AIP and HTTPS traffic. In this situation, the Daemon handles the AIP traffic but forwards the HTTPS traffic on to the SGD web server.

Sometimes the load on the SSL Daemon can affect performance. If you have a multi-processor server, you can tune the number of SSL Daemon processes to the number of processors to improve performance. SSL Daemon tuning is specific to each SGD server. By default, SGD starts four SSL Daemon processes. See [“How to Tune SSL Daemon Processes” on page 38](#) for detail of how to change the number of SSL Daemon processes.

You can use log filters to monitor SSL Daemon processes. By default, all errors are logged. You can increase the amount of log output to assist with tuning or for troubleshooting, see [“How to Change SSL Daemon Log Filters” on page 39](#). The log filters you use have the same format as the log filters used for the SGD server. See [“Using Log Filters to Troubleshoot Problems With an SGD Server” on page 391](#). The same severity and destination file options can be used. By default, all errors are logged to the `/opt/tarantella/var/log` directory.

If the SSL Daemon exits unexpectedly, it makes 10 attempts to restart before failing completely. You can change the maximum number of restart attempts, see [“How to Change SSL Daemon Maximum Restart Attempts” on page 39](#).

▼ How to Tune SSL Daemon Processes

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

- 1. Log in to the SGD host as superuser (root).**

2. Change the number of SSL Daemon processes.

Use the following command:

```
# tarantella config edit \  
--tarantella-config-ssldaemon-minprocesses num \  
--tarantella-config-ssldaemon-maxprocesses num
```

The default *num* is 4.

Use the same value for the minimum and maximum processes.

You tune SSL Daemon processes for the number of processors and *not* for the number of processor cores. Configure no more than one SSL daemon for each processor.

3. Restart the SGD server.

You must restart the SGD server for the change to take effect.

▼ How to Change SSL Daemon Log Filters

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

1. Log in to the SGD host as superuser (root).

2. Change the SSL Daemon log filters.

Use the following command:

```
# tarantella config edit \  
--tarantella-config-ssldaemon-logfilter filter ...
```

Use a comma-separated list of filters.

The default filters are:

```
ssldaemon/*/*error,multi/daemon/*error:sslmulti%%PID%%.log
```

3. Restart the SGD server.

You must restart the SGD server for the change to take effect.

▼ How to Change SSL Daemon Maximum Restart Attempts

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

1. Log in to the SGD host as superuser (root).
2. Change the maximum number of SSL Daemon restart attempts.

Use the following command:

```
# tarantella config edit \  
--tarantella-config-ssldaemon-maxrestarts num
```

The default maximum number is 10. Setting the number of restart attempts to -1 means there is no limit on the number of restart attempts.

3. Restart the SGD server.

You must restart the SGD server for the change to take effect.

Using External SSL Accelerators

SGD supports the use of external SSL accelerators. Performance can be improved by off-loading the processor-intensive transactions required for SSL connections to an external SSL accelerator. External SSL accelerators can also be used to centralize server SSL certificates.

The information in this section applies when an SSL accelerator is used for connections to SGD servers. The *Oracle Secure Global Desktop 4.6 Gateway Administration Guide* has details of how to use SSL accelerators with the SGD Gateway.

To use an external SSL accelerator with SGD, do the following:

- Install the SSL certificate for each SGD server in the array on the external SSL accelerator
- Configure the external SSL accelerator to decrypt SSL connections and forward them as unencrypted connections to SGD
- Enable external SSL accelerator support in SGD

When you enable external SSL accelerator support, the SGD SSL Daemon can accept plain text traffic on the port configured for secure connections, and forward it to SGD as SSL traffic it had decrypted itself.

If you are using server-side proxy servers, you might have to configure your array routes for external SSL accelerators. See [“Configuring Server-Side Proxy Servers” on page 11](#).

▼ How to Enable External SSL Accelerator Support

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

1. **In the Administration Console, go the Secure Global Desktop Servers tab and select an SGD server.**
2. **Go to the Security tab.**
3. **Select the SSL Accelerator Support check box.**
4. **Click Save.**
5. **Restart the SGD server.**

You must restart the SGD server for the change to take effect.

Selecting a Cipher Suite for Secure Connections

You can select the cipher suite that is used for secure connections to SGD servers, see [“How to Change the Cipher Suite for Secure Client Connections” on page 42](#) for details.

A cipher suite is a set of cryptographic algorithms used for the following:

- **Key exchange** – Protects the information required to create shared keys
- **Bulk encryption** – Encrypts messages exchanged between clients and servers
- **Message authentication** – Generates message hashes and signatures to ensure the integrity of a message

A cipher suite specifies one algorithm for each of these tasks. For example, the RSA_WITH_RC4_128_MD5 cipher suite uses RSA for key exchange, RC4 with a 128-bit key for bulk encryption, and MD5 for message authentication.

[TABLE 1-1](#) lists the supported cipher suites.

TABLE 1-1 Supported Cipher Suites for Secure Client Connections

| Supported Cipher Suite | Client Preference | OpenSSL Name |
|---------------------------|-------------------|--------------|
| RSA_WITH_AES_256_CBC_SHA | 1 | AES256-SHA |
| RSA_WITH_AES_128_CBC_SHA | 2 | AES128-SHA |
| RSA_WITH_3DES_EDE_CBC_SHA | 3 | DES-CBC3-SHA |

TABLE 1-1 Supported Cipher Suites for Secure Client Connections

| Supported Cipher Suite | Client Preference | OpenSSL Name |
|------------------------|-------------------|--------------|
| RSA_WITH_RC4_128_SHA | 4 | RC4-SHA |
| RSA_WITH_RC4_128_MD5 | 5 | RC4-MD5 |
| RSA_WITH_DES_CBC_SHA | 6 | DES-CBC-SHA |

When selecting a cipher suite, you use the OpenSSL Name of the cipher suite, as shown in [TABLE 1-1](#). If you select more than one cipher suite, the SGD Client determines which suite is used, based on the client preference order shown in the table above.

By default, the SGD Client uses the RSA_WITH_AES_256_CBC_SHA cipher suite.

▼ How to Change the Cipher Suite for Secure Client Connections

Ensure that no users are logged in to the SGD servers in the array and that there are no running application sessions, including suspended application sessions.

1. **Log in as superuser (root) on the primary SGD server in the array.**
2. **Stop all the SGD servers in the array.**
3. **Specify the cipher suite.**

Use the following command:

```
# tarantella config edit \  
--tarantella-config-security-ciphers cipher-suite ...
```

where *cipher-suite* is the OpenSSL Name of a cipher suite as listed in [“Selecting a Cipher Suite for Secure Connections” on page 41](#).

If you specify more than one *cipher-suite*, use a colon-separated list.

The default setting is AES256-SHA:RC4-MD5

4. **Restart all the SGD servers in the array.**

You must restart the SGD servers for the change to take effect.

Using Connection Definitions

Connection definitions can be used to control whether a secure or a standard connection is used when connecting to an SGD server. The connection type can depend on the following factors:

- The DNS name or IP address of the user's client device
- The SGD server the user logs in to

If SGD security services are not enabled on an SGD server, secure connections to that server are not available regardless of the user's connection definitions.



Caution – If SGD is configured for firewall forwarding, do not use connection definitions. You always use secure connections with firewall forwarding. See [“Firewall Traversal” on page 24](#).

If you use the SGD Gateway, connection definitions are only used for direct client connections that are not routed through an SGD Gateway.

To use connection definitions, you must do the following:

- **Enable connection definition processing** – See [“How to Enable Connection Definition Processing” on page 44](#)
- **Configure connection definitions** – See [“How to Configure Connection Definitions” on page 44](#)

When connection definition processing is enabled, you configure the connection definitions to determine which users receive standard or secure connections. You configure connection definitions at an organization level, which you can override at an organizational unit level or user profile level. By default, all users can receive secure connections if SGD security services are enabled.

Connection definitions use the IP address or DNS name of the client device and the SGD server to determine whether standard or secure connections are used. The order of the connection definitions is important as the first match is used. Connection definitions can include the * or ? wildcards to match more than one DNS name or IP address.

For example, the user profile object for Elizabeth Blue has the following connection definitions:

| Client Device Address | SGD Server Address | Connection Type |
|-----------------------|--------------------|-----------------|
| *.example.com | * | Standard |
| * | * | Secure |

If Elizabeth logs in to SGD from her usual client device, `sales1.example.com`, the first connection definition in the list matches and Elizabeth receives a standard connection.

If Elizabeth logs in to SGD from a client device that is not part of `example.com`, the second connection definition in the list matches and Elizabeth receives a secure connection.

If Elizabeth had no connection definitions, the connection type is determined by the connection definitions of a parent object in the organizational hierarchy.

▼ How to Enable Connection Definition Processing

1. In the Administration Console, go to the Global Settings → Security tab.
2. Select the Connection Definitions check box.
3. Click Save.

▼ How to Configure Connection Definitions

1. In the Administration Console, go to the User Profiles tab and select the object you want to configure.

It is best to configure connection definitions for organization and organizational unit objects as this configures connections definitions for many users at once and makes administration easier.

2. Go to the Security tab.

3. Add a connection definition.

DNS names or IP addresses in a connection definition can include the * or ? wildcards.

- a. In the Connection Definitions table, click the Add button.

The Add New Connection Definition window is displayed.

- b. In the Client Device Address field, type an IP address or DNS name.

- c. In the Secure Global Desktop Server Address, type an IP address or DNS name.

- d. Select a Connection Type from the list.

- e. Click Add.

The Add New Connection Definition window closes and the connection definition is added to the Connection Definitions table.

4. Add further connection definitions as needed.

The Connection Definitions table also shows the definitions that are inherited from parent objects in the organizational hierarchy.

5. Use the Move Up and Move Down buttons to change the order of the connection definitions.

The order of the connection definitions is important. The first matching entry is used. Make sure the most specific definitions appear before more general ones.

User Authentication

This chapter describes how users authenticate to an Oracle Secure Global Desktop (SGD) server to log in to SGD. This is known as *Secure Global Desktop authentication*. Authentication to an application server to run an application, known as *application authentication*, is described in [“Application Authentication” on page 183](#).

This chapter includes the following topics:

- [“Secure Global Desktop Authentication” on page 47](#)
- [“Active Directory Authentication” on page 52](#)
- [“Anonymous User Authentication” on page 61](#)
- [“LDAP Authentication” on page 63](#)
- [“SecurID Authentication” on page 69](#)
- [“Third-Party Authentication” on page 72](#)
- [“UNIX System Authentication” on page 80](#)
- [“Windows Domain Authentication” on page 83](#)
- [“Tuning Directory Services for Authentication” on page 86](#)
- [“Troubleshooting Secure Global Desktop Authentication” on page 102](#)

Secure Global Desktop Authentication

SGD is designed to integrate with your existing authentication infrastructure and has the following two methods for authenticating users:

- **System authentication.** SGD checks the user’s credentials against one or more external authentication services, for example a Lightweight Directory Access Protocol (LDAP) directory. See [“System Authentication Mechanisms” on page 49](#) for details of the available system authentication mechanisms.

- **Third-party authentication.** An external mechanism authenticates the user and SGD trusts that the authentication is correct. The most common use of third-party authentication is web server authentication. See [“Third-Party Authentication” on page 72](#) for more details.

The following are main results of a successful authentication:

- **A user identity.** The name that identifies the user. See [“User Identity” on page 48](#) for more details.
- **A user profile.** The user’s SGD-related settings. See [“User Profile” on page 49](#) for more details.

Sometimes the user identity and the user profile are the same.

In the SGD Administration Console, you can monitor user sessions and application sessions using either the user identity or the user profile.

Depending on how users are authenticated, SGD can prompt users to change their password when they try to log in with an expired password. See [“Password Expiry” on page 50](#) for details.

SGD authentication is global. A user can log in to any SGD server in the array with the same user name and password.

SGD Administrators can enable and disable each authentication mechanism independently, as follows:

- In the Administration Console, use the Global Settings → Secure Global Desktop Authentication tab.
- On the command line, use the `tarantella config` command.

User Identity

A user identity is the name that identify the user. Each authentication mechanism has its own set of rules for determining the user identity.

A user identity is a name assigned by SGD and is sometimes referred to as the *fully qualified name*. The user identity is not necessarily the name of a user profile in the local repository. For example, for LDAP authentication the identity is the distinguished name (DN) of the user in the LDAP directory.

The user identity is associated with the user’s SGD session, their application sessions, and their entries in the application server password cache.

User Profile

A user profile controls a user’s SGD-specific settings. Depending on whether or not you use an LDAP directory to assign applications to users, a user profile can also control the applications a user can access through SGD, sometimes called *webtop content*. Each authentication mechanism has its own set of rules for determining the user profile.

A user profile is always an object in the local repository and is sometimes referred to as an *equivalent name*. A user profile can be a special object called a profile object stored in the System Objects organization. For example, for LDAP authentication the default user profile is o=System Objects/cn=LDAP Profile.

System Authentication Mechanisms

The following table lists the available system authentication mechanisms and describes the basis for authentication.

TABLE 2-1 System Authentication Mechanisms

| Mechanism | Description |
|---|--|
| Anonymous user | Enables users to log in to SGD without using a user name and password. All anonymous users have the same webtop content. See “Anonymous User Authentication” on page 61 . |
| Authentication token | Enables users to log in to SGD if the SGD Client supplies a valid authentication token. Users might have their own webtop content, depending on configuration. Used when the SGD Client operates in Integrated mode, see “Integrated Mode” on page 320 . |
| UNIX system – Search Unix User ID in Local Repository | Enables users to log in to SGD if they have user profiles in the local repository and UNIX® or Linux system accounts on the SGD host. Users might have their own webtop content, depending on configuration. See “UNIX System Authentication” on page 80 . |
| Windows Domain | Enables users to log in to SGD if they belong to a specified Windows domain. Users might have their own webtop content, depending on configuration. See “Windows Domain Authentication” on page 83 |

TABLE 2-1 System Authentication Mechanisms

| Mechanism | Description |
|--|---|
| LDAP | Enables users to log in to SGD if they have an entry in an LDAP directory. Users might have their own webtop content, depending on configuration. See “LDAP Authentication” on page 63 |
| Active Directory | Enables users to log in to SGD if they have an account in an Active Directory forest. Users might have their own webtop content, depending on configuration. See “Active Directory Authentication” on page 52 . |
| UNIX system – Search Unix Group ID in Local Repository | Enables users to log in to SGD if they have UNIX or Linux system accounts on the SGD host. All users in the same UNIX system group have the same webtop content. See “UNIX System Authentication” on page 80 . |
| UNIX system – Use Default User Profile | Enables users to log in to SGD if they have UNIX or Linux system accounts on the SGD host. All UNIX system users have the same webtop content. See “UNIX System Authentication” on page 80 . |
| SecurID | Enables users with RSA SecurID tokens to log in to SGD. Users might have their own webtop content, depending on configuration. See “SecurID Authentication” on page 69 |

When a user logs in, the enabled authentication mechanisms are tried in the order they are listed in [TABLE 2-1](#). When you configure SGD authentication, the Administration Console shows the order in which the mechanisms are tried. The first authentication mechanism that authenticates a user “wins” and no further authentication mechanisms are tried.

Password Expiry

SGD can handle the expiry of the user’s password if configured to do so.

When a user attempts to log in to SGD with an expired password, the Aged Password dialog displays. This dialog does the following:

- Confirms that the password has expired

- Enables the user to enter and confirm a new password

If the new password is accepted, the user is logged in to SGD.

The following table shows which authentication mechanisms support aged passwords.

| Authentication Mechanism | Aged Password Support |
|--|---|
| Active Directory | Yes. See “Configuring SGD for Kerberos Authentication” on page 56 for details. |
| Anonymous user | Not applicable. User logs in without a user name or password. |
| Authentication token | Not applicable. User logs in without a user name or password. |
| LDAP | No. Once a user’s password has expired, they cannot log in to SGD. However, SGD can be configured to force users to change their password before it expires. See “LDAP Bind DN and Password Change” on page 66 for details. |
| SecurID | Yes. If the user’s personal identification number (PIN) has expired, a new PIN dialog is displayed instead of the Aged Password dialog. |
| Third-party (including web server authentication) | No. The expiry of the user’s password is handled by the third-party authentication mechanism and is nothing to do with SGD. |
| UNIX system | Yes. See “UNIX System Authentication and PAM” on page 82 for details. |
| Windows domain | No. |

Security and Passwords

When logging in to SGD, passwords and authentication tokens are only encrypted if there is an Hypertext Transfer Protocol over Secure Socket Layer (HTTPS) connection.

SGD uses external mechanisms for authenticating users. The security of passwords when authenticating users is as follows:

- Active Directory authentication uses the Kerberos protocol for authentication, which is secure
- LDAP authentication can be configured to use a secure connection
- Web server authentication is only secure if the user has an HTTPS connection

- All other authentication mechanisms use the native protocols for authenticating users

Active Directory Authentication

Active Directory authentication enables users to log in to SGD if they have an account in an Active Directory forest. Active Directory authentication offers users a faster, more secure, and more scalable authentication mechanism than LDAP authentication. By using the Kerberos authentication protocol, SGD can securely authenticate any user against any domain in a forest.

Active Directory authentication is disabled by default.

This section includes the following topics:

- [“How Active Directory Authentication Works” on page 52](#)
- [“Setting Up Active Directory Authentication” on page 53](#)
- [“Preparing for Active Directory Authentication” on page 54](#)
- [“Configuring SGD for Kerberos Authentication” on page 56](#)
- [“How to Enable Active Directory Authentication” on page 60](#)

How Active Directory Authentication Works

At the SGD login screen, the user types a user name and password, usually a user name and a domain name joined by the “@” sign, for example `indigo@example.com`.

SGD uses the Kerberos protocol to check the user name and password against a Key Distribution Center (KDC) for a domain.

If the authentication fails, the next authentication mechanism is tried.

If the Kerberos authentication succeeds, SGD establishes the user’s identity by performing an LDAP search of Active Directory. Next, SGD searches for the user profile. See [“User Identity and User Profile” on page 53](#) for details. If the Login attribute of the user profile is not enabled, the user cannot log in and no further authentication mechanisms are tried. If the Login attribute of the user profile is enabled, the user is logged in.

User Identity and User Profile

The user identity is the DN of the LDAP user. In the SGD datastore, the user identity is in the LDAP namespace. In the Administration Console, the text “(LDAP)” is displayed next to the user identity. On the command line, the user identity is located in `.../_service/sco/tta/ldapcache`.

SGD establishes the user profile by searching the local repository, allowing for differences between the LDAP and SGD naming systems. SGD searches for the following until a match is found:

- A user profile with the same name as the user’s LDAP object.
For example, if the user’s LDAP object is `cn=Emma Rald,cn=Sales,dc=example,dc=com`, SGD searches the local repository for `dc=com/dc=example/cn=Sales/cn=Emma Rald`.
- A user profile in the same organizational unit as the user’s LDAP object but with the name `cn=LDAP Profile`.
For example, `dc=com/dc=example/cn=Sales/cn=LDAP Profile`.
- A user profile in any parent organizational unit with the name `cn=LDAP Profile`.
For example, `dc=com/dc=example/cn=LDAP Profile`.

If there is no match, the profile object `o=System Objects/cn=LDAP Profile` is used for the user profile.

You can use Active Directory authentication with Directory Services Integration. The applications assigned to Active Directory users come from a combination of the user profile and from LDAP searches. See [Chapter 3](#) for details of how applications are assigned to users.

Setting Up Active Directory Authentication

Setting up Active Directory authentication involves the following configuration steps:

1. Prepare for Active Directory authentication.
SGD has specific requirements that must be configured before enabling Active Directory authentication, see [“Preparing for Active Directory Authentication” on page 54](#).
2. Configure SGD for Kerberos authentication.
Configure SGD with the details of the KDCs to use for Kerberos authentication, see [“Configuring SGD for Kerberos Authentication” on page 56](#).

3. Enable Active Directory authentication.

Configure SGD to use Active Directory authentication and specify the Active Directory domain details, see [“How to Enable Active Directory Authentication” on page 60](#).

For organizations with complex Active Directory deployments, use service objects to manage and tune your Active Directory configuration. See [“Using Service Objects” on page 90](#).

Preparing for Active Directory Authentication

You prepare for Active Directory authentication as follows:

- Ensure you are using a supported version of Active Directory, see [“Supported Versions of Active Directory” on page 54](#).
- Ensure your Active Directory domains are configured correctly, see [“Domain Requirements” on page 54](#).
- Ensure your SGD servers can connect to Active Directory, see [“Network Requirements for Active Directory Authentication” on page 55](#).
- Synchronize the system clocks, see [“Synchronizing System Clocks” on page 55](#).
- (Optional) Prepare for SSL connections.

SGD connections to Active Directory are always secure. SGD supports two methods for authenticating connections to Active Directory, Kerberos and SSL. Kerberos is the method used by default. To use SSL, additional configuration is required, see [“SSL Connections to Active Directory” on page 56](#).

Supported Versions of Active Directory

The supported versions of Active Directory are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

Domain Requirements

Kerberos authentication must be enabled in Active Directory. It is by default.

Ensure each Active Directory forest has a global catalog server.

When you enable Active Directory authentication, you provide a user name and password. The user must have privileges to search Active Directory for user information. You might want to create a special user reserved for Active Directory authentication.

Network Requirements for Active Directory Authentication

Before you enable Active Directory authentication, make sure all the SGD servers in the array can connect to Active Directory.

SGD must be able to make connections to Active Directory on the following ports:

- Port 53 for DNS lookups on Active Directory
- Ports 88 and 464 for Kerberos authentication to a KDC
- TCP port 389 for the secure LDAP connection to a domain controller
- TCP port 3268 for the secure LDAP connection to a global catalog server

Ports 88 and 464 are the standard ports for Kerberos authentication. These ports are configurable. Port 464 is only required for password change operations. Ports 88 and 464 can use either the TCP or UDP protocol depending on the packet size and your Kerberos configuration, see [“Network Protocols” on page 59](#) for details.

If you are using SSL connections *without client certificates*, SGD must be able to make connections to Active Directory on the following additional ports:

- TCP port 636 for the secure LDAP connection to a domain controller
- TCP port 3269 for the secure LDAP connection to a global catalog server

See [“SSL Connections to Active Directory” on page 56](#) for more details.

SGD performs several DNS lookups to discover LDAP information, see [“Active Directory Authentication and LDAP Discovery” on page 100](#) for details. For these lookups to work, it is essential that your DNS is configured correctly to enable the required information to be returned from Active Directory.

Synchronizing System Clocks

To use Kerberos authentication, the clocks on the KDCs and the SGD servers in the array must be synchronized so that the time is within the maximum tolerance for computer clock synchronization defined for the Kerberos security policy and the Default domain security policy on the Active Directory server. This is called *clock skew*. If the clock skew tolerance is exceeded, the Kerberos authentication fails.

Because time synchronization is important, use Network Time Protocol (NTP) software to synchronize clocks. Alternatively, use the `rdate` command.

SSL Connections to Active Directory

To use SSL for connections to Active Directory, the Active Directory server must be configured to support LDAP signing. Consult your system documentation for details of how to support LDAP signing. [Microsoft KB article 935834](#) has details of how to support LDAP signing for Windows Server 2008.

SGD must be able to validate the SSL certificate presented by an Active Directory server. You might have to import the Certificate Authority (CA) or root certificate for your Active Directory servers into the CA certificates truststore on your SGD servers. See [“The CA Certificate Truststore” on page 408](#) for details of how to check for supported CAs and how to import CA certificates.

By default, SGD authenticates to Active Directory with a user name and password. For additional security, SGD can be configured to present a client certificate instead. If you do this, *each SGD server in the array* must have a valid client certificate that has been signed using Active Directory Certificate Services. Active Directory must be configured to support the use of client certificates.

The process for creating a client certificate is as follows:

1. Create a certificate signing request (CSR) for the client certificate for an SGD server.

See [“How to Create a Client Certificate CSR for an SGD Server” on page 410](#).

2. Create the client certificate using Active Directory Certificate Services.

Consult your system documentation for details of how to create client certificates using Active Directory Certificate Services.

Submit the CSR as a Base-64-encoded certificate request (Advanced Certificate Request).

If you need to select a certificate template for the certificate, the default Administrator or User template is sufficient. The template you select must enable the certificate to be used for user or client authentication.

Ensure you download the client certificate in Base 64 encoded format. If the certificate is signed by an Intermediate CA, download the certificate and chain.

3. Install the client certificate for an SGD server.

See [“How to Install a Client Certificate for an SGD Server” on page 410](#).

Configuring SGD for Kerberos Authentication

To use Active Directory authentication, *every SGD server* in the array must be configured for Kerberos authentication.

A Kerberos configuration file must be present on each SGD server in the array. The Kerberos configuration file used by an SGD server is either of the following:

- **The system default Kerberos configuration file.**

This is usually one the following files:

- `/etc/krb5/krb5.conf` on Solaris OS platforms.
- `/etc/krb5.conf` on Linux platforms.

- **The SGD Kerberos configuration file.**

This is the `/opt/tarantella/bin/jre/lib/security/krb5.conf` file.

You must manually create this file or copy an existing configuration file. If this configuration file exists, it is used *instead of* the system default configuration file.

A Kerberos configuration file contains many options for controlling Kerberos authentication. Consult your system documentation for more details. You might need to configure the following:

- **Kerberos realms and KDCs.** The KDCs SGD uses to authenticate users, see [“Kerberos Realms and KDCs” on page 57](#)
- **Password expiry.** Whether or not SGD prompts a user for a new password if their password has expired, see [“Active Directory Password Expiry” on page 58](#).
- **Network protocol.** Whether SGD uses the User Datagram Protocol (UDP) or Transmission Control Protocol (TCP) for Kerberos authentication, see [“Network Protocols” on page 59](#)
- **KDC timeout.** What happens if there is a failure to contact a KDC, see [“KDC Timeout” on page 59](#)



Caution – Pay particular attention to the format of the `krb5.conf` file. Incorrectly formatted files can cause problems, especially with password expiry operations.

Whenever you make changes to your Kerberos configuration file, SGD does not detect the changes until you restart the SGD server. Alternatively, you can use the following commands to refresh the Kerberos configuration and connection information without restarting the SGD server:

```
$ tarantella cache --flush krb5config
$ tarantella cache --flush ldapconn
```

Kerberos Realms and KDCs

As a minimum, the Kerberos configuration file must contain the following sections:

- [libdefaults]. This sets defaults for Kerberos authentication. You must set the `default_realm`. If a default realm is not specified, users might not be able to change an expired password.
- [realms]. This sets the KDCs for each Kerberos realm. A realm can have more than one KDC. The entry for each KDC has the form *host:port*. The *port* can be omitted if the default port 88 is used.
- [domain_realm]. This maps network domains to Kerberos realms.

The following is an example Kerberos configuration file:

```
[libdefaults]
default_realm = EXAMPLE.COM

[realms]
EXAMPLE.COM = {
    kdc = kdc.example.com
}
EAST.EXAMPLE.COM = {
    kdc = ad01.east.example.com
    kdc = ad02.east.example.com
}
WEST.EXAMPLE.COM = {
    kdc = ad01.west.example.com
    kdc = ad02.west.example.com
}

[domain_realm]
example.com = EXAMPLE.COM
.east.example.com = EAST.EXAMPLE.COM
east.example.com = EAST.EXAMPLE.COM
.west.example.com = WEST.EXAMPLE.COM
west.example.com = WEST.EXAMPLE.COM
```

Active Directory Password Expiry

SGD can be configured to prompt a user for a new password if their Active Directory password has expired. If a default realm is not specified in the `krb5.conf` file, users are unable to change an expired password.

To configure password expiry, the details of the server that handles the password change *for each Kerberos realm* must be added to the Kerberos configuration file, as follows:

```
kpasswd_server = host:port
admin_server = host:port
kpasswd_protocol = SET_CHANGE
```

The `kpasswd_server` and `admin_server` lines identify the Kerberos administration server that handles the password change. If `kpasswd_server` is omitted, the `admin_server` is used instead. The *port* can be omitted if the default port 464 is used.

The following is an example of password expiry configuration for a realm:

```
EAST.EXAMPLE.COM = {  
    kdc = ad01.east.example.com  
    kdc = ad02.east.example.com  
    admin_server = ad01.east.example.com  
    kpasswd_protocol = SET_CHANGE  
}
```

SGD can be configured to warn users that their password is about to expire and force them to change their password, see [“Password Expiry” on page 95](#). SGD can only do this if it can read the domain controller’s Maximum Password Age setting and the user’s Password Last Set attribute. If you configure SGD to search only the global catalog, this attribute is not available. See [“Search Only the Global Catalog” on page 97](#) for more details.

Network Protocols

When sending messages to the KDC or the Kerberos administration server, SGD uses either the UDP or TCP protocols. The protocol used is determined by the following line in the `[libdefaults]` section of the Kerberos configuration file:

```
udp_preference_limit = bytes
```

This line sets the maximum size in bytes for packets that can be sent using UDP. If the message is larger than this size, TCP is used. If the KDC or administration server indicates that the package is too big, TCP is used instead. To always use TCP, set the `udp_preference_limit` as follows:

```
udp_preference_limit = 1
```

KDC Timeout

You can configure a KDC timeout that controls how long SGD waits for a reply from a KDC, and how many times it tries to contact each KDC.

To set the KDC timeout, add the following lines to the [libdefaults] section of the Kerberos configuration file:

```
kdc_timeout = time
max_retries = number
```

The `kdc_timeout` sets the maximum number of milliseconds to wait for a reply from a KDC. The `max_retries` is the maximum number of times each KDC is tried. The KDCs for each realm are tried in the order they are listed in the [realms] section of the Kerberos configuration file.

It is best to keep the KDC timeout and the LDAP discovery timeout in step. If you increase the KDC timeout, increase the LDAP discovery timeout. See [“LDAP Discovery Timeout” on page 90](#).

If SGD cannot contact any KDCs for the user’s realm, the authentication phase fails.

▼ How to Enable Active Directory Authentication

1. **In the Administration Console, display the Secure Global Desktop Authentication Configuration Wizard.**

Go to the Global Settings → Secure Global Desktop Authentication tab and click the Change Secure Global Desktop Authentication button.

2. **On the Third-Party/System Authentication step, ensure the System Authentication check box is selected.**
3. **On the System Authentication - Repositories step, select the LDAP/Active Directory check box.**
4. **On the LDAP Repository Details step, configure the Active Directory forest details.**

- a. **For Repository Type, select the Active Directory option.**

- b. **In the URLs field, type the URL of an Active Directory forest.**

For example, `ad://example.com`.

The URL must start with `ad://`. Only type one URL

- c. **Configure secure connections to Active Directory.**

- **To use only the Kerberos protocol for secure connections** – Select the Kerberos option for Connection Security, and type a user name and password in the User Name and Password fields. This option is selected by default.

- **To use Kerberos and SSL for secure connections** – Select the SSL option for Connection Security, and type a user name and password in the User Name and Password fields.
- **To use Kerberos, SSL, and client certificates for secure connections** – Select the SSL option for Connection Security, and select the Use Certificates check box.

See [“SSL Connections to Active Directory” on page 56](#) for details of the additional configuration required to use SSL connections.

If you type a user name, the user name has the form `user@example.com`. If you omit the domain name from the user name, SGD uses the information in the URL, Active Directory Base Domain, and Active Directory Default Domain fields to obtain a domain. The user must have privileges to search Active Directory for user information.

d. (Optional) In the Active Directory Base Domain field, type a partial domain name.

The base domain is used when users only supply a partial domain when they log in. For example, if the base domain is set to `example.com` and a user logs in with the user name `rouge@west`, SGD authenticates the user as `rouge@west.example.com`.

e. (Optional) In the Active Directory Default Domain field, type a domain name to use as a default.

The default domain is used when users do not supply a domain when they log in. For example, if the default domain is set to `east.example.com` and a user logs in with the user name `rouge`, SGD authenticates the user as `rouge@east.example.com`.

5. On the Review Selections step, check the authentication configuration and click Finish.

When you click Finish, SGD creates a service object called `generated`. Service objects are used to manage directory services configuration, see [“Using Service Objects” on page 90](#) for more details.

Anonymous User Authentication

Anonymous user authentication enables users to log in to SGD without using a user name and password.

As users are anonymous, SGD assigns each anonymous user a temporary user identity. The user identity is only effective while the user is logged in.

Anonymous user authentication is disabled by default.

This section includes the following topics:

- [“How Anonymous User Authentication Works” on page 62](#)
- [“How to Enable Anonymous User Authentication” on page 63](#)

How Anonymous User Authentication Works

At the SGD login screen, the user clicks the Log In button, leaving the user name and password blank.

If the user types a user name or a password, the authentication fails and the next authentication mechanism is tried.

If both the user name and the password are blank, the user is authenticated and is logged in.

User Identity and User Profile

As the user does not supply a user name or password when they log in, SGD assigns a temporary user identity. In the SGD datastore, the user identity is in the DNS namespace. In the Administration Console, the user identity is displayed as `server:number (anon)`. On the command line, the user identity is displayed as `.../_dns/server/_anon/number`.

The profile object `System Objects/Anonymous Profile` is always used for the user profile. All anonymous users receive the same webtop content.

Application Sessions and Password Cache Entries

Each user logged in anonymously has independent application sessions. The application sessions end automatically when the user logs out *even if* the application is configured to be always resumable.

All password cache entries belong to the `System Objects/Anonymous User Profile` object. All anonymous users share the same application server passwords. Anonymous users cannot add or change entries in the password cache. This means that, unless an SGD Administrator has cached application server passwords for them, anonymous users are prompted for a password *every time* they start an application. Use the Administration Console or the `tarantella passcache` command to manage application server passwords for the `System Objects/Anonymous User Profile` object.

▼ How to Enable Anonymous User Authentication

1. In the Administration Console, display the Secure Global Desktop Authentication Configuration Wizard.

Go to the Global Settings → Secure Global Desktop Authentication tab and click the Change Secure Global Desktop Authentication button.

2. On the Third-Party/System Authentication step, ensure the System Authentication check box is selected.
3. On the System Authentication - Repositories step, select the Anonymous check box.
4. On the Review Selections step, check the authentication configuration and click Finish.

LDAP Authentication

LDAP authentication enables users to log in to SGD if they have an entry in an LDAP directory.

This authentication mechanism is disabled by default.

This section includes the following topics:

- [“How LDAP Authentication Works” on page 63](#)
- [“Setting Up LDAP Authentication” on page 65](#)
- [“Preparing for LDAP Authentication” on page 65](#)
- [“How to Enable LDAP Authentication” on page 67](#)

How LDAP Authentication Works

At the SGD login screen, the user types a user name and password. The user name can be any of the following:

- A common name, for example Indigo Jones
- A user name, for example indigo
- An email address, for example indigo@example.com

SGD searches the LDAP directory for an LDAP object with an attribute that matches the user name typed by the user. By default, SGD searches the following attributes:

- cn

- uid
- mail

If an LDAP object is not found, the next authentication mechanism is tried.

If an LDAP object is found, SGD performs a bind using the name of the LDAP object and the password typed by the user. If the bind fails, the next authentication mechanism is tried.

If the authentication succeeds, SGD searches the local repository for the user profile, see [“User Identity and User Profile” on page 64](#) for details. If the Login attribute of the user profile is not enabled, the user cannot log in and no further authentication mechanisms are tried. If the Login attribute of the user profile is enabled, the user is logged in.

User Identity and User Profile

The user identity is the DN of the user’s LDAP object. In the SGD datastore, the user identity is in the LDAP namespace. In the Administration Console, the text “(LDAP)” is displayed next to the user identity. On the command line, the user identity is located in `.../_service/sco/tta/ldapcache`.

SGD establishes the user profile by searching the local repository, allowing for differences between the LDAP and SGD naming systems. SGD searches for the following until a match is found:

- A user profile with the same name as the user’s LDAP object.
For example, if the user’s LDAP object is `cn=Emma Rald,cn=Sales,dc=example,dc=com`, SGD searches the local repository for `dc=com/dc=example/cn=Sales/cn=Emma Rald`.
- A user profile in the same organizational unit as the user’s LDAP object but with the name `cn=LDAP Profile`.
For example, `dc=com/dc=example/cn=Sales/cn=LDAP Profile`.
- A user profile in any parent organizational unit with the name `cn=LDAP Profile`.
For example, `dc=com/dc=example/cn=LDAP Profile`.

If there is no match, the profile object `o=System Objects/cn=LDAP Profile` is used for the user profile.

You can use LDAP authentication with Directory Services Integration. The applications assigned to LDAP users come from a combination of the user profile and from LDAP searches. See [Chapter 3](#) for details of how applications are assigned to users.

Setting Up LDAP Authentication

Setting up LDAP authentication involves the following configuration steps:

1. Prepare for LDAP authentication.

Additional configuration might be required to use SGD with your LDAP directory, see [“Preparing for LDAP Authentication” on page 65](#).

2. Enable LDAP authentication.

Configure SGD to use LDAP authentication and specify the LDAP directory details, see [“How to Enable LDAP Authentication” on page 67](#).

For organizations with complex LDAP deployments, use service objects to manage and tune your LDAP configuration. See [“Using Service Objects” on page 90](#).

Preparing for LDAP Authentication

You prepare for LDAP authentication as follows:

- Ensure you are using a supported LDAP directory, see [“Supported LDAP Directories” on page 65](#).
- Ensure your SGD servers can connect to the LDAP directory servers, see [“Network Requirements for LDAP Authentication” on page 65](#).
- Ensure that you understand the requirements for the LDAP bind DN and SGD password change operations, see [“LDAP Bind DN and Password Change” on page 66](#)
- Ensure that Novell eDirectory users can authenticate, see [“Authentication to Novell eDirectory” on page 67](#)

Supported LDAP Directories

The supported LDAP directories are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

Network Requirements for LDAP Authentication

Before you enable LDAP authentication, make sure all the SGD servers in the array can contact each LDAP directory server used for authentication.

The standard port used for connections to LDAP directory servers is TCP port 389 for standard connections and port TCP 636 for secure (`ldaps://`) connections. If your directory servers use different ports, you can specify the port when you enable LDAP authentication. You must make sure SGD can make LDAP connections using these ports.

To be able to use secure (`ldaps://`) connections, SGD must be able to validate the SSL certificate presented by an LDAP directory server. You might have to import the CA certificates for your LDAP directory servers into the SGD CA certificate truststore. See [“The CA Certificate Truststore” on page 408](#) for details of how to check for supported CAs and how to import CA certificates.

LDAP Bind DN and Password Change

By default, SGD uses two LDAP bind DN, an administrator bind DN and a user bind DN.

The administrator bind DN is the user name and password configured for LDAP authentication. The administrator bind DN is used only for querying the directory server and so this user must have privileges to search the directory. You might want to create a special LDAP user for use with SGD. The administrator bind can be an anonymous bind. Active Directory does not support anonymous binds.

The user bind DN is the user name and password provided when a user logs in. By default, the user bind DN is used for authentication and password change operations.

Once a user’s password expires, they cannot log in to SGD and SGD cannot force them to change their password. SGD can be configured to warn users that their password is about to expire, and to force them to change their password before it expires, see [“Password Expiry” on page 95](#). For SGD to be able to do this, the following must be true:

- On LDAP directories, SGD must be able to read the user’s password policy control when binding to the directory as the user
- On Active Directory, SGD must be able to read the domain controller’s Maximum Password Age setting and the user’s Password Last Set attribute

If your directory server does not meet these requirements, and you want SGD to handle password change, you must configure SGD to use the administrator bind DN for password change operations. See [“LDAP Password Update Mode” on page 96](#).

Note – On some LDAP directories, password change operations performed using the administrator bind DN are treated as a password reset rather than a change operation.

For *Oracle Directory Server Enterprise Edition*, if you configure SGD to use the administrator bind DN for password updates, additional configuration might be needed for SGD to handle password change, as follows:

- Do not use the “User must change password after reset” option either in the global password policy or for an individual password policy. This causes the password change to fail.
- The administrator bind DN must have administrative privileges.

For *Active Directory*, password expiry including forcing the user to change their password at next logon, can only be handled if there is a secure connection (`ldaps://`) between the SGD server and the Active Directory server.

By default, *Novell eDirectory* requires that all simple LDAP binds that contain a password must use an SSL connection. To use eDirectory with SGD, do either of the following:

- Configure SGD to use secure connections to eDirectory by using `ldaps://` URLs
- Configure the LDAP group object in eDirectory and disable TLS for simple binds

Authentication to Novell eDirectory

Users might not be able to authenticate Novell eDirectory because the user login filter for LDAP authentication filters for the `cn` attribute and this attribute is a restricted attribute in eDirectory.

To enable users to log in to SGD, do either of the following:

- Ensure that the LDAP bind DN has privileges to access to the `cn` attribute. See [“LDAP Bind DN and Password Change” on page 66](#) for more details.
- Change the user login filter so that it does not filter the `cn` attribute. See [“Filtering LDAP or Active Directory Logins” on page 87](#) for more details.

▼ How to Enable LDAP Authentication

1. **In the SGD Administration Console, display the Secure Global Desktop Authentication Configuration Wizard.**

Go to the Global Settings → Secure Global Desktop Authentication tab and click the Change Secure Global Desktop Authentication button.

2. **On the Third-Party/System Authentication step, ensure the System Authentication check box is selected.**
3. **On the System Authentication - Repositories step, select the LDAP/Active Directory check box.**

4. On the LDAP Repository Details step, configure the LDAP directory details.

a. For Repository Type, select the LDAP option.

Select this option even if you are using a Microsoft Active Directory server for LDAP authentication. The Active Directory option enables Active Directory authentication, see [“Active Directory Authentication” on page 52](#).

b. In the URLs field, type the URL of one or more LDAP directory servers.

For example, `ldap://melbourne.example.com`.

If you type more than one URL, separate each URL with a semicolon (;).

If there is than one URL, SGD uses the URLs in the order they are listed. If the first LDAP directory server in the list is unavailable, the next one is tried.

To use secure connections to LDAP directory servers, use an `ldaps://` URL.

The URLs must all be of the same type, either `ldap://` or `ldaps://`. You cannot use a mixture of `ldap://` and `ldaps://` URLs.

If the LDAP directory uses a non-standard port, specify the port number as part of the URL, for example `ldap://melbourne.example.com:5678`. Otherwise the port number can be omitted.

You can specify a DN to use as the search base, for example `ldap://melbourne.example.com/dc=example,dc=com`. This specifies the part of the LDAP directory used to search for the user identity.

c. Type the details of an LDAP user in the User Name and Password fields.

The user name must be the DN of the user, for example `cn=sgd-user,cn=Users,dc=example,dc=com`. This is the administrator bind DN, see [“LDAP Bind DN and Password Change” on page 66](#) for more details.

As you can only enter one user name and password, this user must be able to search all LDAP directory servers listed in the URL field.

If the directory server supports anonymous binds, you can omit the user name and password. You must be able to perform LDAP queries for user data to use anonymous binds.

The URLs configured for an LDAP service object must all be of the same type, either `ldap://` or `ldaps://`. You cannot use a mixture of `ldap://` and `ldaps://` URLs.

5. On the Review Selections step, check the authentication configuration and click Finish.

When you click Finish, SGD creates a service object called `generated`. Service objects are used to manage directory services configuration, see [“Using Service Objects” on page 90](#) for more details.

SecurID Authentication

SecurID authentication enables users with RSA SecurID tokens to log in to SGD. SGD authenticates users against an RSA Authentication Manager, formerly known as ACE/Server.

RSA SecurID is a product from RSA Security, Inc., that uses two-factor authentication based on something you *know*, a PIN, and something you *have*, a tokencode supplied by a separate token such as a PIN pad, standard card, or software token. The PIN and tokencode are combined to form a passcode which is used as the password when you log in to SGD.

This authentication mechanism is disabled by default.

This section includes the following topics:

- [“How SecurID Authentication Works” on page 69](#)
- [“Setting Up SecurID Authentication” on page 70](#)
- [“Configuring SGD Servers as Agent Hosts” on page 71](#)
- [“How to Enable SecurID Authentication” on page 72](#)

How SecurID Authentication Works

At the SGD login screen, the user types their SecurID user name, for example `indigo`, and their passcode.

This authentication mechanism searches the local repository for a user profile with a Name attribute that matches the user name typed by the user. If there is no match, the search is repeated on the Login Name attribute, and finally on the Email Address attribute.

If a user profile is found, the Login Name attribute of that object is used as the SecurID user name. If no user profile is found, the name the user typed is used as the SecurID user name.

Next, SGD checks the SecurID user name, and the passcode typed by the user, against the RSA Authentication Manager. If the authentication fails, the user cannot log in because there are no further authentication mechanisms to try.

If the authentication succeeds and the Login attribute for the user profile is not enabled, the user is not logged in. If the authentication succeeds and the Login attribute for the user profile is enabled, the user is logged in.

User Identity and User Profile

If a user profile was found in the local repository, this is used for the user identity and user profile. In the SGD datastore, the user identity is in the Local namespace. In the Administration Console, the text “(Local)” is displayed next to the user identity. On the command line, the user identity is located in `.../_ens`.

If no user profile was found in the local repository, the user identity is the SecurID user name. In the SGD datastore, the user identity is in the SecurID namespace. In the Administration Console, the text “(SecurID)” is displayed next to the user identity. On the command line, the user identity is located in `.../_service/sco/tta/securid`.

The profile object `System Objects/SecurID User Profile` is used for the user profile.

Setting Up SecurID Authentication

Setting up SecurID authentication involves the following configuration steps:

1. Install and configure RSA SecurID.

Ensure you are using a supported version of RSA SecurID. The supported versions of SecurID are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

Ensure the RSA Authentication Manager is up to date with the latest patches released by RSA.

2. Configure each SGD server in the array as an Agent Host.

Each SGD server in the array acts as an Agent Host so that it can authenticate users against the RSA Authentication Manager.

See “[Configuring SGD Servers as Agent Hosts](#)” on page 71.

3. Enable SecurID authentication in SGD.

Configure SecurID authentication so that SecurID users can log in to SGD.

See “[How to Enable SecurID Authentication](#)” on page 72.

Configuring SGD Servers as Agent Hosts

To use SecurID authentication, *each SGD server* in the array must be configured as an Agent Host. As SecurID implementations can vary, the following procedure is an example only. Consult your SecurID documentation for details of how to configure an Agent Host.

▼ How to Configure an SGD Server as an Agent Host

Before you begin, ensure you have access to the RSA Authentication Manager configuration file, `sdconf.rec`.

1. Log in as superuser (root) on the SGD host.
2. Ensure the SGD server can contact the RSA Authentication Manager on the network.

You might have to open ports in your firewalls to enable an SGD server to contact the RSA Authentication Manager.

The default ports that must be open are the following:

- UDP port 5500 from the SGD server to the Authentication Manager.
- UDP ports 1024 to 65535 from the Authentication Manager to the SGD server.

3. Specify the location of the RSA Authentication Manager configuration file.

- a. Create the `/etc/sdace.txt` file with the following content:

```
VAR_ACE=/opt/ace/data
```

- b. Save the file.

4. Copy the RSA Authentication Manager configuration file to the SGD server.

- a. Create an `/opt/ace/data` directory.
- b. Copy the `sdconf.rec` file to the `/opt/ace/data` directory.

5. Set the file permissions so that SGD can read and write the configuration files.

```
# chmod 444 /etc/sdace.txt
# chown -R ttasys:ttaserv /opt/ace
# chmod -R 775 /opt/ace
```

6. Register the SGD servers as Agent Hosts in the RSA Authentication Manager database.

Use either the RSA Authentication Manager Database Administration application or `sdadmin` application.

Add the SGD server as a UNIX Agent Host in the database, using the fully qualified name, `server.domain.com`.

For each Agent Host, Configure Group or User Activation. Alternatively, set the Open to All Locally Known Users option.

▼ How to Enable SecurID Authentication

1. In the SGD Administration Console, display the Secure Global Desktop Authentication Configuration Wizard.

Go to the Global Settings → Secure Global Desktop Authentication tab and click the Change Secure Global Desktop Authentication button.

2. On the Third-Party/System Authentication step, ensure the System Authentication check box is selected.

3. On the System Authentication - Repositories step, select the SecurID check box.

4. On the Review Selections step, check your authentication configuration and click Finish.

Third-Party Authentication

Third-party authentication enables users to log in to SGD if they have been authenticated by an external mechanism.

If you are using the SGD webtop, the only form of third-party authentication you can use is web server authentication. If you develop your own webtop applications using SGD web services, you can use any third-party authentication mechanism.

Third-party authentication is disabled by default.

This section includes the following topics:

- [“How Third-Party Authentication Works” on page 73](#)
- [“Setting Up Third-Party Authentication” on page 75](#)
- [“How to Enable Third-Party Authentication” on page 76](#)

- [“SGD Administrators and Third-Party Authentication” on page 77](#)
- [“Trusted Users and Third-Party Authentication” on page 77](#)

How Third-Party Authentication Works

The user types in a user name and password directly to the external mechanism, typically using a web browser’s authentication dialog.

Third-party authentication is based on trust. SGD trusts that the third-party mechanism has authenticated the user correctly and so they are authenticated to SGD.

Next SGD performs a search to establish the user identity and user profile. The following search methods can be used:

- Search Local Repository
- Search LDAP Repository
- Use Default Third-Party Identity

If more than one search method is enabled, the methods are tried in the order they are listed above. The first matching user identity found is used. The search methods are described in the following sections.

If the searches do not produce a match, SGD cannot establish an identity for the user and the user cannot log in. If you are using the SGD webtop, the standard login page displays so that the user can log in using system authentication.

Search Local Repository

The Search Local Repository method searches the local repository for a user profile with a Name attribute that matches the user’s third-party user name. If there is no match, the search is repeated on the Login Name attribute, and finally on the Email Address attribute. If no user profile is found, the next search method is tried.

User Identity and User Profile

If a user profile is found, that object is used for the user identity and user profile. In the SGD datastore, the user identity is in the Local namespace. In the Administration Console, the text “(Local)” is displayed next to the user identity. On the command line, the user identity is located in `.../_ens`.

Search LDAP Repository

The Search LDAP Repository method searches an LDAP directory for an LDAP object with an attribute that matches the user name typed by the user. By default, SGD searches the following attributes:

- cn
- uid
- mail

If an LDAP object is found, the password typed by the user is checked against the LDAP object. If the authentication fails, the next authentication mechanism is tried.

If an LDAP object is not found, the next search method is tried.

User Identity and User Profile

The user identity is the DN of the user's LDAP object. In the SGD datastore, the user identity is in the LDAP namespace. In the Administration Console, the text "(LDAP)" is displayed next to the user identity. On the command line, the user identity is located in `.../_service/sco/tta/ldapcache`.

Next SGD searches for the user profile. When searching for the user profile, you can specify Use Default LDAP Profile or Use Closest Matching LDAP Profile. Use Default LDAP Profile is the default.

If Use Default LDAP Profile is selected, the profile object `o=System Objects/cn=LDAP Profile` is used for the user profile.

If Use Closest Matching LDAP Profile is selected, SGD establishes the user profile by searching the local repository, allowing for differences between the LDAP and SGD naming systems. SGD searches for the following until a match is found:

- A user profile with the same name as the user's LDAP object.
For example, if the LDAP object is `cn=Emma Rald,cn=Sales,dc=example,dc=com`, SGD searches the local repository for `dc=com/dc=example/cn=Sales/cn=Emma Rald`.
- A user profile in the same organizational unit as the user's LDAP object but with the name `cn=LDAP Profile`.
For example, `dc=com/dc=example/cn=Sales/cn=LDAP Profile`.
- A user profile in any parent organizational unit with the name `cn=LDAP Profile`.
For example, `dc=com/dc=example/cn=LDAP Profile`.

If there is no match, the profile object `o=System Objects/cn=LDAP Profile` is used for the user profile.

Use Default Third-Party Identity

The Use Default Third-Party Identity method does not perform a search.

User Identity and User Profile

The user identity is the third-party user name. In the SGD datastore, the user identity is in the Third-party namespace. In the Administration Console, the text “(3rd party)” is displayed next to the user identity. On the command line, the user identity is located in `.../_service/sco/tta/thirdparty`.

The profile object `System Objects/Third Party Profile` is always used for the user profile.

Setting Up Third-Party Authentication

Setting up third-party authentication involves the following configuration steps:

1. (Optional) Prepare to use LDAP directories.

Third-party authentication can be configured to search an LDAP directory to establish users' identities.

Check that you are using a supported LDAP directory, see [“Supported LDAP Directories” on page 65](#).

Additional configuration might be required to use SGD with your LDAP directory, see [“Preparing for LDAP Authentication” on page 65](#).

For organizations with complex LDAP deployments, use service objects to manage and tune your LDAP configuration. See [“Using Service Objects” on page 90](#).

2. Enable third-party authentication.

See [“How to Enable Third-Party Authentication” on page 76](#).

By default, SGD Administrators cannot log in to SGD when third-party authentication is enabled, see [“SGD Administrators and Third-Party Authentication” on page 77](#).

3. Enable a third-party authentication mechanism.

The most common authentication mechanism used with third-party authentication is web server authentication.

▼ How to Enable Third-Party Authentication

1. **In the SGD Administration Console, display the Secure Global Desktop Authentication Configuration Wizard.**

Go to the Global Settings → Secure Global Desktop Authentication tab and click the Change Secure Global Desktop Authentication button.

2. **On the Third-Party/System Authentication step, select the Third-Party Authentication check box.**
3. **On the Third-Party Authentication - User Identity and Profile step, select the check box for one or more search methods for finding the user identity.**

For details on how the search methods work, see [“How Third-Party Authentication Works” on page 73](#).

If the Search LDAP Repository check box is selected, select an option for finding the LDAP user profile.

4. **(Optional) On the LDAP Repository Details step, configure the LDAP directory details.**

The LDAP Repository Details step only displays if an LDAP search method is selected in Step 3.

- a. **For Repository Type, select the LDAP option.**

Select this option even if you are using a Microsoft Active Directory server for LDAP authentication. The Active Directory option enables Active Directory authentication, see [“Active Directory Authentication” on page 52](#).

- b. **In the URLs field, type the URL of one or more LDAP directory servers.**

For example, `ldap://melbourne.example.com`.

If you type more than one URL, separate each URL with a semicolon (;).

SGD uses the URLs in the order they are listed. If the first LDAP directory server in the list is unavailable, the next one is tried.

To use secure connections to LDAP directory servers, use an `ldaps://` URL.

The URLs must all be of the same type, either `ldap://` or `ldaps://`. You cannot use a mixture of `ldap://` and `ldaps://` URLs.

If the LDAP directory uses a non-standard port, specify the port number as part of the URL, for example `ldap://melbourne.example.com:5678`.

Otherwise the port number can be omitted.

You can specify a DN to use as the search base, for example

`ldap://melbourne.example.com/dc=example,dc=com`. This specifies the part of the LDAP directory used to search for the user identity.

c. Type the details of an LDAP user in the User Name and Password fields.

The user name must be the DN of the user, for example `cn=sgd-user, cn=Users, dc=example, dc=com`. This is the administrator bind DN, see [“LDAP Bind DN and Password Change” on page 66](#) for more details.

As you can only enter one user name and password, this user must be able to search all LDAP directory servers listed in the URL field.

If the directory server supports anonymous binds, you can omit the user name and password. You must be able to perform LDAP queries for user data to use anonymous binds.

5. On the Review Selections step, check the authentication configuration and click Finish.

If you enabled the LDAP search method, SGD creates a service object called `generated`. Service objects are used to manage directory services configuration, see [“Using Service Objects” on page 90](#) for more details.

SGD Administrators and Third-Party Authentication

By default, third-party authentication does not permit SGD Administrators to log in to SGD. This is a security measure. To change this behavior, use the following command:

```
$ tarantella config edit \  
--tarantella-config-login-thirdparty-allowadmins 1
```

Trusted Users and Third-Party Authentication

Third-party authentication gives users access to SGD *without* having to authenticate to an SGD server. SGD is able to trust the third-party authentication mechanism because client applications, such as the webtop, and the SGD server have a shared secret which is the user name and password of a trusted user.

In a standard installation, there is just one trusted user. However, you might want to create additional trusted users in the following circumstances:

- You relocate the webtop to a different JavaServer Pages™ (JSP™) technology container on a different host.
- You develop your own client applications, using the SGD `com.tarantella.tta.webservices.client.views` package, either on the same host as SGD or on a different host.

- You have concerns about the security of the default trusted user.

You create and maintain the “database” of trusted users on each SGD server in the array. The database is not shared between SGD servers. See [“How to Create a New Trusted User” on page 79](#) for details of how to add a trusted user. Note the following:

- The `tarantella webservice add_trusted_user` command is the only supported way to store trusted users on the SGD server.
- To change the password of an existing trusted user, you must first delete the user with the `tarantella webservice delete_trusted_user` command and then create the user again.
- Every time you make a change to a trusted user, you must restart the SGD web server.
- Usually client applications only use the credentials of a single trusted user to access SGD services.

Information for Application Developers

If you are using SGD web services to develop your own applications, the `ITarantellaExternalAuth` web service is used for third-party authentication. This web service is protected with Basic web server authentication so that you can only access it using the credentials of a trusted user. This is configured as follows:

- The `http://SGD-server/axis/services/document/externalauth` URL is protected in the configuration file for the Axis web application
`/opt/tarantella/webserver/tomcat/tomcat-version/webapps/axis/WEB-INF/web.xml`
- The Tomcat component of the SGD web server is configured to support Basic web server authentication using Tomcat’s `MemoryRealm` and `Secure Hash Algorithm (SHA)` digested passwords. This is in the Tomcat configuration file
`/opt/tarantella/webserver/tomcat/tomcat-version/conf/server.xml`.
- The list of trusted users is stored in the Tomcat users configuration file
`/opt/tarantella/webserver/tomcat/tomcat-version/conf/tomcat-users.xml`

If you have developed your own client applications using the `com.tarantella.tta.webservices.client.views` package, you can store the trusted user credentials for the application in the same way as the webtop, see [“How to Create a New Trusted User” on page 79](#). Otherwise, you need to develop your own methods for storing the credentials.

▼ How to Create a New Trusted User

Repeat the following procedure on each SGD server in the array.

1. Log in as superuser (root) on the SGD host.
2. Stop the SGD web server.
3. Add the new trusted user to the database of trusted users on the SGD server.
 - a. Think of a user name and password for the trusted user.
 - b. Create the trusted user.

Use the following command:

```
# tarantella webserver add_trusted_user username
```

When prompted, type the password.

- c. Check the user is created.

Use the following command:

```
# tarantella webserver list_trusted_users
```

- d. Check that the trusted user works.

Go to the `http://SGD-server/axis/services/document/externalauth` URL. When prompted, log in as the trusted user.

4. Add the new trusted user to the web services resources file for the webtop application.

If you have relocated the webtop to a different host, perform this step on the remote host.

- a. Encode the user name and password of the trusted user.

Use the following command:

```
# /opt/tarantella/bin/jre/bin/java -classpath \  
/opt/tarantella/webserver/tomcat/tomcat-version/webapps/sgd/WEB-INF/lib/sgd-  
webservices.jar \  
com.tarantella.tta.webservices.client.views.SgdPasswd \  
--encode username:password
```

- b. Copy the encoded user name and password from the output.

c. Change to the shared resources directory.

```
# cd /opt/tarantella/webserver/tomcat/tomcat-version
# cd shared/classes/com/tarantella/tta/webservices/client/views
```

d. Edit the `Resources.properties` file.

e. Replace the text after `sgdaccess=` with the encoded user name and password.

f. Save the changes.

5. Start the SGD web server.

UNIX System Authentication

UNIX system authentication enables users to log in to SGD if they have UNIX or Linux system accounts on the SGD host.

UNIX system authentication is enabled by default.

This section includes the following topics:

- [“How UNIX System Authentication Works” on page 80](#)
- [“UNIX System Authentication and PAM” on page 82](#)
- [“How to Enable UNIX System Authentication” on page 83](#)

How UNIX System Authentication Works

UNIX system authentication supports the following search methods for authenticating users against a UNIX or Linux system user database and determining the user identity and profile:

- Search Unix User ID in Local Repository
- Search Unix Group ID in Local Repository
- Use Default User Profile

These search methods are described in the following sections.

Search Unix User ID in Local Repository

At the SGD login screen, the user types a user name and password. The user name can be any of the following:

- A common name, for example Indigo Jones
- A user name, for example indigo
- An email address, for example indigo@example.com

SGD searches the local repository for a user profile with a Name attribute that matches what the user typed. If there is no match, the search is repeated on the Login Name attribute, and finally on the Email Address attribute. If no user profile is found, the next authentication mechanism is tried.

If a user profile is found, the Login Name attribute of that object is treated as a UNIX or Linux system user name. This user name, and the password typed by the user, are checked against the UNIX or Linux system user database. If the authentication fails, the next authentication mechanism is tried.

If the authentication succeeds and the Login attribute for the user profile is not enabled, the user is not logged in and no further authentication mechanisms are tried. If the authentication succeeds and the Login attribute for the user profile is enabled, the user is logged in.

This search method is enabled by default.

User Identity and User Profile

The matching user profile in the local repository is used for the user identity and user profile. In the SGD datastore, the user identity is in the Local namespace. In the Administration Console, the text “(Local)” is displayed next to the user identity. On the command line, the user identity is located in `.../_ens`.

Search Unix Group ID in Local Repository

SGD checks the user name and password typed by the user at the login screen against the UNIX or Linux system user database.

If the authentication fails, the next authentication mechanism is tried.

If the authentication succeeds, SGD searches for the user profile. See [“User Identity and User Profile” on page 82](#) for details. If the Login attribute of the user profile object is not enabled, the user cannot log in and no further authentication mechanisms are tried. If the Login attribute of the user profile is enabled, the user is logged in.

This search method is enabled by default.

User Identity and User Profile

The user identity is the UNIX or Linux system user name. In the SGD datastore, the user identity is in the User namespace. In the Administration Console, the text “(UNIX)” is displayed next to the user identity. On the command line, the user identity is located in `.../_user`.

SGD searches the local repository for a user profile `cn=gid`, where *gid* is the UNIX system group ID of the authenticated user. If found, this is used as the user profile. If the user belongs to more than one group, the user’s primary or effective group is used. If no user profile is found in the local repository, the profile object `System Objects/UNIX User Profile` is used for the user profile.

Use Default User Profile

SGD checks the user name and password typed by the user at the login screen against the UNIX or Linux system user database.

If the authentication fails, the next authentication mechanism is tried.

If the authentication succeeds, the user is logged in.

This search method is disabled by default.

User Identity and User Profile

The user identity is the UNIX or Linux system user name. In the SGD datastore, the user identity is in the User namespace. In the Administration Console, the text “(UNIX)” is displayed next to the user identity. On the command line, the user identity is located in `.../_user`.

The profile object `System Objects/UNIX User Profile` is used for the user profile. All UNIX system users receive the same webtop content.

UNIX System Authentication and PAM

SGD supports Pluggable Authentication Modules (PAM). UNIX system authentication uses PAM for user authentication, account operations, and password operations.

When you install SGD on Linux platforms, the SGD installation program automatically creates PAM configuration entries for SGD by copying the current configuration for the `passwd` program and creating the `/etc/pam.d/tarantella` file.

When you install SGD on Solaris OS platforms, you must add PAM configuration entries manually. For example, you might add these entries for `tarantella` to the `/etc/pam.conf` file.

```
tarantella auth required pam_unix_auth.so.1
tarantella password required pam_unix_auth.so.1
```

▼ How to Enable UNIX System Authentication

1. **In the SGD Administration Console, display the Secure Global Desktop Authentication Configuration Wizard.**

Go to the Global Settings → Secure Global Desktop Authentication tab and click the Change Secure Global Desktop Authentication button.

2. **On the Third-Party/System Authentication step, ensure the System Authentication check box is selected.**
3. **On the System Authentication - Repositories step, select the Unix check box.**
4. **On the Unix Authentication - User Profile step, select the check box for one or more search methods for finding the user profile.**
See [“How UNIX System Authentication Works” on page 80](#) for details on the search methods.
5. **On the Review Selections step, check the authentication configuration and click Finish.**

Windows Domain Authentication

Windows domain authentication enables users to log in to SGD if they belong to a specified Windows domain.

Windows domain authentication is disabled by default.

This section includes the following topics:

- [“How Windows Domain Authentication Works” on page 84](#)
- [“How to Enable Windows Domain Authentication” on page 85](#)

- “Passwords, Domains, and Domain Controllers” on page 85

How Windows Domain Authentication Works

At the SGD login screen, the user types a user name and password. The user name can be any of the following:

- A common name, for example Indigo Jones
- A user name, for example indigo
- An email address, for example indigo@example.com

SGD searches the local repository for a user profile with a Name attribute that matches the user name typed by the user. If there is no match, the search is repeated on the Login Name attribute, and finally on the Email Address attribute.

If a user profile is found, the Login Name attribute of the user profile is treated as the Windows domain user name. If no user profile is found, the name the user typed is used as the Windows domain user name. SGD then checks the Windows domain user name and the password typed by the user against the domain controller.

If the authentication fails, the next authentication mechanism is tried.

If the authentication succeeds and the Login attribute for the user profile is not enabled, the user is not logged in and no further authentication mechanisms are tried.

If the authentication succeeds and either the Login attribute for the user profile is enabled or no matching user profile is found, the user is logged in.

User Identity and User Profile

If a user profile was found in the local repository, that object is used for the user identity and user profile. In the SGD datastore, the user identity is in the Local namespace. In the Administration Console, the text “(Local)” is displayed next to the user identity. On the command line, the user identity is located in `.../_ens`.

If no user profile was found in the local repository, the user identity is the Windows domain user name. In the SGD datastore, the user identity is in the NT namespace. In the Administration Console, the text “(NT)” is displayed next to the user identity. On the command line, the user identity is located in `.../_service/sco/tta/ntauth`. The profile object `System Objects/NT User Profile` is used for the user profile.

▼ How to Enable Windows Domain Authentication

1. **In the SGD Administration Console, display the Secure Global Desktop Authentication Configuration Wizard.**
Go to the Global Settings → Secure Global Desktop Authentication tab and click the Change Secure Global Desktop Authentication button.
2. **On the Third-Party/System Authentication step, ensure the System Authentication check box is selected.**
3. **On the System Authentication - Repositories step, select the Windows Domain Controller check box.**
4. **On the Windows Domain Authentication - Domain Controller step, type the name of a domain controller in the Windows Domain field.**
5. **On the Review Selections step, check the authentication configuration and click Finish.**

Passwords, Domains, and Domain Controllers

Windows domain authentication supports 8-bit case-sensitive passwords. The user name can contain any characters.

If you need to authenticate users from more than one domain, you must have one domain that is trusted by all the other domains. You must use the trusted domain as the Windows domain controller when you configure Windows domain authentication.

When a user from another domain logs in to SGD, they must use the format `domain\username` for their user name. If they do not use this format, SGD tries to authenticate the user using the authentication domain and fails.

Note – The Domain Name (`--ntdomain`) attribute for user profiles plays no part in the SGD login.

If an SGD server is on a different subnet to the domain controller, you must hard code the domain controller information, see [“How to Specify a Domain Controller on a Different Subnet”](#) on page 86.

▼ How to Specify a Domain Controller on a Different Subnet

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

1. Log in a superuser (root) on the SGD host.
2. Stop the SGD server.
3. Configure the domain controller

Use the following commands:

```
# tarantella config edit \  
--com.sco.tta.server.login.ntauth.NTAuthService.properties-authConfig \  
authnbt=NTNAME  
# tarantella config edit \  
--com.sco.tta.server.login.ntauth.NTAuthService.properties-authConfig-append \  
authserver=my.domain.name
```

where *NTNAME* is the NetBIOS name of the domain controller and *my.domain.name* is the DNS name or IP address of the domain controller.

4. Start the SGD server.

Tuning Directory Services for Authentication

This section describes how to tune directory services configuration and includes the following topics:

- [“Filtering LDAP or Active Directory Logins” on page 87](#)
- [“LDAP Discovery Timeout” on page 90](#)
- [“Using Service Objects” on page 90](#)
- [“Password Expiry” on page 95](#)
- [“LDAP Password Update Mode” on page 96](#)
- [“Whitelists” on page 97](#)
- [“Sites” on page 96](#)
- [“Search Only the Global Catalog” on page 97](#)
- [“Suffix Mappings” on page 98](#)

- “Domain Lists” on page 99
- “Lookup Cache Timeout” on page 99
- “LDAP Operation Timeout” on page 99
- “Active Directory Authentication and LDAP Discovery” on page 100

Filtering LDAP or Active Directory Logins

You can use login filters to control which users can log in to SGD and to specify which attributes are used to identify users. There are two filters, user login filter and group login filter.

You might want to configure a login filter in the following circumstances:

- **Users unable to log in.** If your directory uses particular attributes for identifying users, users might not be able to log in to SGD. The solution is to configure the user login filter SGD to search for additional attributes.
- **Slow login times.** SGD checks for several attributes when identifying users and this can lead to slow login times if you have a large directory. You can improve login times by changing the user login filter to check for a smaller number of attributes.
- **Restrict user logins.** If you want to restrict which users can log in to SGD, configure the group login filter or the user login filter.

User Login Filter

Whenever SGD searches a directory to establish a user’s identity, it uses a user login filter to check for attributes on the object in the directory.

For LDAP and third-party authentication, the following login filter is used:

```
( | (cn=${name}) (uid=${name}) (mail=${name})) )
```

For Active Directory authentication, the following login filter is used:

```
( | (cn=${user}) (sAMAccountName=${user}) (userPrincipalName=${user}@${domain}) (userPrincipalName=${name}) (mail=${name})) )
```

These login filters contain the following variables:

- `${name}` – The full name typed by the user in the SGD login page
- `${user}` – The part of the user name before the “@” symbol
- `${domain}` – The part of the user name after the “@” symbol. This can be generated by SGD using the base domain, default domain, or suffix mapping settings for the service object

See [“How to Configure a User Login Filter” on page 88](#) for details of how to change the user login filter.

Group Login Filter

The group search filter is used to restrict the LDAP or Active Directory users that can log in to SGD, by testing the user’s group membership. If the user is not a member of the group, the user cannot log in to SGD.

See [“How to Configure the Group Login Filter” on page 89](#) for details of how to apply the filter to SGD.

▼ How to Configure a User Login Filter

Ensure that no users are logged in to the primary SGD server in the array and that there are no running application sessions, including suspended application sessions.

1. Log in as superuser (root) on the primary SGD server in the array.
2. Stop the SGD server.
3. Configure a user login filter.

Caution – Any mistakes in this step can result in users being unable to log in.

- For LDAP authentication and third-party authentication, use the following command:

```
# tarantella config edit \  
--com.sco.tta.server.login.ldap.LdapLoginAuthority.properties-searchFilter filter
```


- For Active Directory authentication, use the following command:

```
# tarantella config edit \  
--com.sco.tta.server.login.ADLoginAuthority.properties-searchFilter filter
```

For example, to configure a user login filter to search for just the `uid` and `mail` attributes, use the following filter:

```
" (& (| (uid=\${name}) (mail=\${name}))) "
```

For example, to configure a user login filter to search for just the `cn` and `mail` attributes, and to permit logins for only for users that are in the sales department, use the following filter:

```
" (& (| (cn=\${name}) (mail=\${name})) (department=sales)) "
```

If you use variables in the filter, you must backslash escape the “\$” symbol.

4. Start the SGD server.

▼ How to Configure the Group Login Filter

Ensure that no users are logged in to the primary SGD server in the array and that there are no running application sessions, including suspended application sessions.

1. Log in as superuser (root) on the primary SGD server in the array.
2. Stop the SGD server.
3. Configure the group login filter.

Caution – Any mistakes in this step can result in users being unable to log in.

To configure the group login filter, use the following command:

```
# tarantella config edit \  
--com.sco.tta.server.login.DSLoginFilter.properties-loginGroups group_dn ...
```

where *group_dn* is the DN of one or more LDAP groups.

For example, to permit logins only from users who are members of either the `cn=sgdusers` or the `cn=sgdadmins` group, use the following command:

```
# tarantella config edit \  
--com.sco.tta.server.login.DSLoginFilter.properties-loginGroups \  
"cn=sgdusers,cn=groups,dc=example,dc=com" \  
"cn=sgdadmins,cn=groups,dc=example,dc=com"
```

4. Start the SGD server.

LDAP Discovery Timeout

The LDAP discovery timeout controls how long SGD waits for a directory server (LDAP or Active Directory) to respond to the initial contact request. The default is 20 seconds.

SGD makes two attempts to contact a directory server. If there is no response, SGD tries another directory server. If all directory servers time out, SGD might not be able to authenticate users or assign applications to them.

To change the timeout, use the following command:

```
$ tarantella config edit --tarantella-config-ldap-timeout secs
```

For Active Directory authentication, the LDAP discovery timeout must be longer than the KDC timeout. See [“KDC Timeout” on page 59](#). For example, if the KDC timeout is 10 seconds and the KDC retries is 3, make the LDAP discovery timeout 35 seconds (3 x 10 seconds + extra 5 seconds). Keep the KDC timeout and the LDAP discovery timeout in step. If you increase the KDC timeout, increase the LDAP discovery timeout.

Using Service Objects

A service object is a group of directory services configuration settings used for the following SGD authentication mechanisms:

- Active Directory authentication, see [“Active Directory Authentication” on page 52](#)
- LDAP authentication, see [“LDAP Authentication” on page 63](#)
- Third-party authentication using the LDAP repository search, see [“Third-Party Authentication” on page 72](#)

You can enable Active Directory authentication or LDAP authentication. Both cannot be enabled at the same time.

If you enable any of these authentication mechanisms using the Authentication Wizard on the Global Settings → Secure Global Desktop Authentication tab in the SGD Administration Console, SGD automatically creates a service object called `generated`. If you enable any of these authentication methods on the command line, you must manually create at least one service object.

You can create and manage service objects on the Global Settings → Service Objects tab in the SGD Administration Console. The Service Objects List table shows the available service objects for the SGD array.

On the command line, you use the `tarantella service` command to create and manage service objects.

Note the following points about service objects:

- **Service objects have a type.** This is either an LDAP type or an Active Directory type. The type controls which SGD authentication mechanism can use the object. Active Directory service objects are used only for Active Directory authentication.
- **Service objects can be enabled or disabled.** A service object must be enabled before it can be used for authentication. Typically you disable a service object only because you know a directory service is temporarily unavailable, or because you know it will become available in the future.
- **Service objects have a position.** SGD uses the enabled service objects in the order specified in the Service Objects List table. If the first enabled service object in the list does not result in a successful authentication or user identity, the next enabled service object in the list is tried.

In the Administration Console, you can configure only the commonly-used settings for service objects, such as the URL of the directory server, or the user name and password. See [“How to Create an Active Directory Service Object” on page 92](#) and [“How to Create an LDAP Service Object” on page 93](#) for more details.

There are also some advanced service object settings that can be configured only from the command line with the `tarantella service new` or the `tarantella service edit` commands.

For LDAP service object types, the following are the advanced configuration settings:

- [“Password Expiry” on page 95](#)
- [“LDAP Password Update Mode” on page 96](#)
- [“Lookup Cache Timeout” on page 99](#)
- [“LDAP Operation Timeout” on page 99](#)

For AD service object types, the following are the advanced configuration settings:

- [“Password Expiry” on page 95](#)
- [“Sites” on page 96](#)
- [“Whitelists” on page 97](#)
- [“Blacklists” on page 97](#)
- [“Search Only the Global Catalog” on page 97](#)
- [“Suffix Mappings” on page 98](#)
- [“Domain Lists” on page 99](#)
- [“Lookup Cache Timeout” on page 99](#)
- [“LDAP Operation Timeout” on page 99](#)

▼ How to Create an Active Directory Service Object

1. **In the Administration Console, display the Global Settings → Service Objects tab.**

2. **In the Service Objects List table, click the New button.**

The Create New Service Object window is displayed.

3. **In the Name field, type the name of the service object.**

Once you have created a service object, you cannot rename it. The name can only contain lowercase characters, digits, or the characters “_” and “-”.

4. **For Type, select the Active Directory option.**

Once you have created a service object, you cannot change the type.

5. **(Optional) Deselect the Enabled check box.**

A service object must be enabled before SGD can use it. Only enable a service object if you are sure that it is ready to be used.

6. **In the URLs field, type the URL of an Active Directory forest.**

For example, `ad://example.com`.

The URL must start with `ad://`. Only type one URL.

The URL must be unique. Different service objects cannot use the same URL.

Use the Test button to test the connection to the URL.

7. **Configure secure connections to Active Directory.**

- **To use only the Kerberos protocol for secure connections** – Select the Kerberos option for Connection Security, and type a user name and password in the User Name and Password fields.

Note – The Kerberos option is selected by default.

- **To use Kerberos and SSL for secure connections** – Select the SSL option for Connection Security, and type a user name and password in the User Name and Password fields.

- **To use Kerberos, SSL, and client certificates for secure connections** – Select the SSL option for Connection Security, and select the Use Certificates check box.

See [“SSL Connections to Active Directory” on page 56](#) for details of the additional configuration required to use SSL connections.

If you type a user name, the user name has the form `user@example.com`. If you omit the domain name from the user name, SGD uses the information in the URL, Base Domain, and Default Domain fields to obtain a domain. The user must have privileges to search Active Directory for user information.

8. (Optional) In the Active Directory Base Domain field, type a partial domain name.

The base domain is used when users only supply a partial domain when they log in. For example, if the base domain is set to `example.com` and a user logs in with the user name `rouge@west`, SGD tries to authenticate the user as `rouge@west.example.com`.

9. (Optional) In the Active Directory Default Domain field, type a domain name.

The default domain is used when users do not supply a domain when they log in. For example, if the default domain is set to `east.example.com` and a user logs in with the user name `rouge`, the SGD tries to authenticate the user as `rouge@east.example.com`.

10. Click the Create button.

The Create New Service Object window is closed and an entry for the service object is added at the bottom of the Service Objects List table in last position on the Service Objects tab.

11. Use the Move Up and Move Down buttons to change the position of the service object in the table.

SGD uses the enabled service objects in the order they are shown.

▼ How to Create an LDAP Service Object

1. In the Administration Console, display the Global Settings → Service Objects tab.

2. In the Service Objects List table, click the New button.

The Create New Service Object window is displayed.

3. In the name field, type the name of the service object.

Once you have created a service object, you cannot rename it. The name can only contain lowercase characters, digits, or the characters “_” and “-”.

4. For Type, select the LDAP option.

Select this option even if you are using a Microsoft Active Directory server for LDAP authentication.

Once you have created a service object, you cannot change the type.

5. (Optional) Deselect the Enabled check box.

A service object must be enabled before SGD uses it. Only enable a service object if you are sure that it is ready to be used.

6. In the URLs field, type the URL of one or more LDAP directory servers.

For example, `ldap://melbourne.example.com`.

If you type more than one URL, separate each URL with a semicolon (;).

SGD uses the URLs in the order they are listed. If the first LDAP directory server in the list is unavailable, the next one is tried.

To use secure connections to LDAP directory servers, use an `ldaps://` URL.

The URLs configured for an LDAP service object must all be of the same type, either `ldap://` or `ldaps://`. You cannot use a mixture of `ldap://` and `ldaps://` URLs.

If the LDAP directory uses a non-standard port, specify the port number as part of the URL, for example `ldap://melbourne.example.com:5678`. Otherwise the port number can be omitted.

You can specify a DN to use as the search base, for example `ldap://melbourne.example.com/dc=example,dc=com`. This specifies the part of the LDAP directory used to search for the user identity.

The URL(s) must be unique. Different service objects cannot use the same URL(s).

Use the Test button to test the connection to the URL(s).

7. Type the details of an LDAP user in the User Name and Password fields.

The user name must be the DN of the user, for example `cn=sgd-user,cn=Users,dc=example,dc=com`. This is the administrator bind DN, see [“LDAP Bind DN and Password Change” on page 66](#) for more details.

As you can only enter one user name and password, this user must be able to search all LDAP directory servers listed in the URL field.

If the directory server supports anonymous binds, you can omit the user name and password. You must be able to perform LDAP queries for user data to use anonymous binds.

8. Click the Create button.

The Create New Service Object window is closed and an entry for the service object is added at the bottom of the Service Objects List table in last position on the Service Objects tab.

9. Use the Move Up and Move Down buttons to change the position of the service object in the table.

SGD uses the enabled service objects in the order they are shown.

Password Expiry

The following information applies to LDAP and Active Directory service object types. See [“Using Service Objects” on page 90](#).

SGD can handle the expiry of a user’s password in the following ways:

- Display a warning message on the webtop, telling the user that their password is about to expire
- Deny authentication and force the user to reset their password at the next log in

The password expiry features are disabled by default.

For important information about authentication and password expiry, see [“Active Directory Password Expiry” on page 58](#) and [“LDAP Bind DN and Password Change” on page 66](#).

Password expiry features are configured separately for each service object. For example, to enable LDAP password expiry features for the `mainldap` service object, use the following command:

```
$ tarantella service edit --name mainldap --check-pwd-policy 1
```

With this configuration, users see a warning message on their webtop seven days before their password is due to expire. One day before their password is due to expire, SGD forces them to change their password. These are the defaults.

You can control when the password expiry features become active. In the following example, the password expiry features are enabled for the `mainldap` service object. SGD is configured to display a warning message 14 days (1209600 seconds) before password expiry, and users are forced to change their passwords 3 days (259200 seconds) before they expire:

```
$ tarantella service edit --name mainldap --check-pwd-policy 1 \  
--pwd-expiry-warn-threshold 1209600 \  
--pwd-expiry-fail-threshold 259200
```

LDAP Password Update Mode

The following information applies only to LDAP service object types. See [“Using Service Objects” on page 90](#).

By default, SGD performs password changes by performing a bind as the LDAP user. See [“LDAP Bind DN and Password Change” on page 66](#) for more details.

The password update mode can be configured separately for each LDAP service object. For example, to use the administrator bind for the mainldap service object, use the following command:

```
$ tarantella service edit --name mainldap
--password-update-mode ldapadmin
```

Sites

The following information applies only to Active Directory service object types. See [“Using Service Objects” on page 90](#).

A site object in Active Directory is an object that contains subnets.

If site awareness is enabled for a service object, SGD automatically attempts to discover site information by contacting the global catalog. Alternatively, you can configure your own site name for the service object.

If SGD has site information, it queries only the Active Directory servers appropriate for the site.



Caution – If you configure a whitelist, the site configuration for the service object is ignored, see [“Whitelists” on page 97](#) for more details.

See [“Active Directory Authentication and LDAP Discovery” on page 100](#) for more details about how sites are used.

Sites can be configured separately for each service object. For example, to enable site awareness for the east service object, use the following command:

```
$ tarantella service edit --name east --site-aware 1
```

For example, to configure a site name of boston for the east service object, use the following command:

```
$ tarantella service edit --name east --site-aware 1 --site-name boston
```


Whitelists

The following information applies only to Active Directory service object types. See [“Using Service Objects” on page 90](#).

A whitelist is a list of global catalog servers that are *always* used for LDAP queries. Only those servers that are included in the whitelist can be used for LDAP queries.



Caution – If you configure a whitelist, the site configuration for the service object is ignored, see [“Sites” on page 96](#) for more details.

The order of the servers in the whitelist is important. If SGD cannot contact the first server in the list, it tries the next one.

See [“Active Directory Authentication and LDAP Discovery” on page 100](#) for more details about how whitelists are used.

Whitelists can be configured separately for each service object. For example, to configure a whitelist for the `east` service object, use the following command:

```
$ tarantella service edit --name east --white-list \  
"good1.east.example.com" "good2.east.example.com"
```

Blacklists

The following information applies only to Active Directory service object types. See [“Using Service Objects” on page 90](#).

A blacklist is a list of Active Directory servers that are *never* used for LDAP queries, for example because a server is temporarily unavailable. Blacklists override any other configuration such as sites or whitelists.

Blacklists can be configured separately for each service object. For example, to configure a blacklist for the `east` service object, use the following command:

```
$ tarantella service edit --name east --black-list \  
"bad1.east.example.com" "bad2.east.example.com"
```

Search Only the Global Catalog

The following information applies only to Active Directory service object types. See [“Using Service Objects” on page 90](#).

When searching for user information from Active Directory, SGD uses the domain controller for the user's domain by default.

While the domain controller contains the most complete and reliable source for user information, it can result in longer timeouts and delays because SGD has to manage connections to both the domain controller and the global catalog. Instead SGD can be configured to search for user information only from the global catalog.



Caution – If you enable this option, you cannot use the service object password expiry options because SGD cannot access the user's Password Last Set attribute. See [“Password Expiry” on page 95](#) for more details. Also access to group information is reduced because SGD cannot access domain local security group information.

See [“Active Directory Authentication and LDAP Discovery” on page 100](#) for more details about how this option is used.

Whether to search only the global catalog can be configured separately for each for service object. For example, to enable searching only the global catalog for the `east` service object, use the following command:

```
$ tarantella service edit --name east --ad-alwaysusegc 1
```

After running this command, you must flush the cache of LDAP data. Run the following command as superuser (root) on *every SGD server in the array*:

```
# tarantella cache --flush all
```

Suffix Mappings

The following information applies only to Active Directory service object types. See [“Using Service Objects” on page 90](#).

Suffix mappings enable you to remap the domain typed by a user to an authentication domain.

Suffix mappings can be configured separately for each service object. For example, to configure a suffix mapping for the `east` service object, use the following command:

```
$ tarantella service edit --name east --suffix-mappings \  
"test.east.example.com=east.example.com"
```

Each suffix mapping has the format `suffix=domain`. If there is more than one mapping, separate each mapping with a space.

Domain Lists

The following information applies only to Active Directory service object types. See [“Using Service Objects” on page 90](#).

When SGD starts, it connects to the configured Active Directory forests and then only connects to individual domains as they are needed. In some situations, there can be delays for users when they log in while SGD establishes a connection to the domain. Performance can be improved by providing SGD with a list of domains to connect to when SGD starts.

Domain lists can be configured separately for each for service object. For example, to configure a domain list for the `east` service object, use the following command:

```
$ tarantella service edit --name east --domain-list \
  "boston.east.example.com" "cambridge.east.example.com"
```

If there is more than one domain, separate each domain name with a space.

Domain lists do not restrict the Active Directory domains used for authentication.

Lookup Cache Timeout

The following information applies to LDAP and Active Directory service object types. See [“Using Service Objects” on page 90](#).

The lookup cache timeout controls for how long SGD keeps LDAP user data after a user logs in. The default is 1200 seconds (20 minutes).

You might want to increase this timeout if the LDAP data does not change very often.

The lookup cache timeout can be configured separately for each for service object. For example, to configure the lookup cache timeout for the `east` service object, use the following command:

```
$ tarantella service edit --name east --lookupcache-timeout secs
```

LDAP Operation Timeout

The following information applies to LDAP and Active Directory service object types. See [“Using Service Objects” on page 90](#).

You can configure an LDAP timeout in the event that the LDAP searches of an LDAP directory or Active Directory server takes too long. The LDAP timeout controls how long SGD waits for the directory server to respond to LDAP operations, such as requests for data. The default is 20 seconds.

SGD makes two attempts to contact the directory server. If there is no response, SGD tries another directory server in the list. For Active Directory service objects, see [“Active Directory Authentication and LDAP Discovery” on page 100](#) for details of the servers SGD contacts. For LDAP service objects, the list of LDAP servers comes from the URLs configured for the service object.

If all directory servers time out, SGD might not be able to authenticate users, or assign applications using LDAP.

The LDAP operation timeout can be configured separately for each service object. For example, to configure the timeout for the `east` service object, use the following command:

```
$ tarantella service edit --name east --operation-timeout secs
```

Active Directory Authentication and LDAP Discovery

With Active Directory authentication, once a user has been authenticated using Kerberos, SGD performs an LDAP search of Active Directory to establish the user identity and other user information. By default, SGD performs the following steps to discover the LDAP information:

1. **DNS lookup for global catalogs.** SGD performs a DNS lookup using the URL configured for the service object to obtain a list of global catalog servers for the forest.
2. **LDAP query on a global catalog.** SGD performs an LDAP query on a global catalog to establish the user identity.

SGD queries the global catalog servers in the order they are returned from the DNS lookup. If SGD cannot contact the first global catalog, it tries the next one in the list.

3. **DNS lookup for domain controllers.** SGD performs a DNS lookup for the domain controllers for the user’s domain.

The domain used for this lookup is either the domain typed by the user when they log in, or constructed using the default domain and base domain configured for the service object.

4. **LDAP query on a domain controller.** SGD performs an LDAP query on a domain controller to establish a complete set of attributes for the user, such as group membership.

SGD queries the domain controllers in the order they are returned from the DNS lookup. If SGD cannot contact the first domain controller, it tries the next one in the list.

The configuration of the service object can have a significant effect on the discovery of LDAP information, see [“Using Service Objects” on page 90](#) for more details. The follow table summarizes the effect of service objects on the steps performed by SGD.

| Service Object Configuration Options | Steps Performed | Notes |
|--|-----------------|---|
| Whitelist | 2, 3, 4 | The LDAP query is only on the global catalogs in the whitelist. |
| Search global catalog | 1, 2 | The DNS lookup and LDAP query are only on the global catalogs. No operations performed on the domain controllers. |
| Whitelist Search global catalog | 2 | The LDAP query is only on the global catalogs in the whitelist. No operations performed on the domain controllers. |
| Site aware Site name | 1, 2, 3, 4 | The DNS lookups are only on the global catalogs and domain controllers for the configured site. |
| Site aware Site name Search global catalog | 1, 2 | The DNS lookup is only on the global catalogs for the configured site. No operations performed on the domain controllers. |
| Site aware | 1, 2, 3, 4 | SGD performs an additional DNS lookup to obtain a list of global catalogs, and then performs an LDAP ping to a global catalog to discover the site name. The DNS lookups are only for the global catalogs and domain controllers for the discovered site. |
| Site aware Search global catalog | 1, 2 | SGD performs an additional DNS lookup to obtain a list of global catalogs, and then performs an LDAP ping to a global catalog to discover the site name. The DNS lookup is only for the global catalogs for the discovered site. No operations performed on the domain controllers. |

Troubleshooting Secure Global Desktop Authentication

Use the information in this section to troubleshoot problems when users log in to SGD. This section includes the following topics:

- [“Setting Log Filters for Authentication Problems” on page 102](#)
- [“Denying Users Access to SGD After Failed Login Attempts” on page 103](#)
- [“Users Cannot Log In to Any SGD Server” on page 104](#)
- [“Using Shared Accounts for Guest Users” on page 105](#)
- [“Solaris OS Users Cannot Log in When Security is Enabled” on page 105](#)
- [“An Ambiguous User Name Dialog Is Displayed When a User Tries to Log in” on page 106](#)

Setting Log Filters for Authentication Problems

To help diagnose problems with Secure Global Desktop authentication, use one or more of the log filters shown in the following table to obtain more information.

| Log Filter | Purpose |
|----------------------------|--|
| server/directoryservices/* | Information about authentication mechanisms that use directory services. Applies to Active Directory, LDAP, and third-party authentication. |
| server/login/* | Information about what happens when users attempt to log in. Applies to all authentication mechanisms. |
| server/securid/* | Information about connections to RSA Authentication Manager. Applies to SecurID authentication. |

For information about setting log filters, see [“Using Log Filters to Troubleshoot Problems With an SGD Server” on page 391](#).

Denying Users Access to SGD After Failed Login Attempts

SGD Administrators can enable a login failure handler so that users are denied access to SGD after three failed login attempts. See [“How to Enable the Login Failure Handler” on page 103](#). This additional security measure only works if users have their own user profile objects in the local repository. It does not work for the default profile objects in the System Objects organization. See for details

The number of login attempts is configurable, see [“How to Change the Number of Login Attempts” on page 103](#). By default users get three attempts. The number of login attempts is local to each SGD server and is not copied across the array. Only when the login limit is reached on a server, is the user denied access across the array. For example, a user could try to log in on each SGD server two times, but only when they fail for the third time on a server are they denied access to the other members of the array.

If a user is denied access, they are only denied access to SGD. They are not denied access to the host on which SGD is installed

When a user is denied access, SGD deselects the Login check box on the General tab (`--enabled false`) for the user profile object in the Administration Console. To give a user access again, you must select the check box (`--enabled true`).

For security reasons, users are not given any indication that their account is disabled. They see the same message as if they had typed an incorrect password.

▼ How to Enable the Login Failure Handler

You can only enable the login failure handler from the command line.

- Use the following command:

```
$ tarantella config edit \  
--tarantella-config-components-loginfailurehandler 1 \  
--tarantella-config-components-loginfailurefilter 1
```

▼ How to Change the Number of Login Attempts

Ensure that no users are logged in to the SGD servers in the array and that there are no running application sessions, including suspended application sessions.

1. Log in to the primary SGD server as superuser (root).
2. Stop the primary SGD server.

3. Set the number of login attempts.

Use the following command:

```
# tarantella config edit \  
--com.sco.tta.server.login.LoginFailureHandler.properties-attemptsallowed num
```

4. Start the primary SGD server.

5. Do a warm restart of all secondary SGD servers.

Use the following command:

```
# tarantella restart sgd --warm
```

Users Cannot Log In to Any SGD Server

If all users, including the UNIX system root user, cannot log in to any SGD server, this might be caused by either of the following:

- All authentication mechanisms are disabled
- User logins to all SGD servers are disabled

To check whether all authentication mechanisms are disabled, use the following command:

```
$ tarantella config list | grep login
```

If all authentication mechanisms are disabled, enable the UNIX system authentication mechanism from the command line, as follows:

```
$ tarantella config edit --login-ens 1
```

Once the UNIX system authentication mechanism is enabled, you can log in to the Administration Console with the user name “Administrator” and the UNIX system root user’s password. You can then reconfigure authentication.

To check whether user logins are disabled for an SGD server, use the following command:

```
$ tarantella config list --server serv... --server-login
```

If user logins to all SGD servers are disabled, use the following command to enable user logins:

```
$ tarantella config edit --array --server-login 1
```


Using Shared Accounts for Guest Users

SGD enables more than one user to log in using the same user name and password, for example to share an account for guest users.

Note – Anonymous users are always treated as using a shared account, see [“Anonymous User Authentication” on page 61](#).

Users that share a user profile object share the same application server passwords. Guest users cannot add or change entries in the password cache. This means that, unless an SGD Administrator has cached application server passwords for them, guest users are prompted for a password *every time* they start an application. Use the Administration Console or the `tarantella passcache` command to manage application server passwords for guest users.

▼ How to Share a User Profile Between Users

1. In the Administration Console, go to the User Profiles tab.
2. Select the user profile that is to be shared.
The General tab is displayed.
3. For Login, select the Multiple check box.
4. Click Save.

Solaris OS Users Cannot Log in When Security is Enabled

If users with Solaris OS client devices find that they cannot log in to an SGD server when SGD security services are enabled, check that the `/dev/random` device is present on the client device.

SGD security services require the `/dev/random` device. If it is missing, install the Solaris OS patch that contains this device.

An Ambiguous User Name Dialog Is Displayed When a User Tries to Log in

The Ambiguous User Name dialog is displayed only for users who share person object attributes and also have the same password.

For example, there are two users with the name John Smith (cn=John Smith) and they have chosen the same password. Their email addresses and user names are different. If they log in with the name John Smith, SGD displays the Ambiguous User Name dialog which asks them to provide either an email address or a user name. The dialog displays because the credentials they supply match more than one user. If they log in using an email address or a user name, they are logged in.

The Ambiguous User Name dialog is displayed only if you are using LDAP authentication or UNIX system authentication that searches for the user ID in the local repository.

The solution is to ensure that users have unique passwords. Alternatively, configure the user profiles to have unique attributes. SGD uses the Name (`--name`), Login Name (`--user`) and Email Address (`--email`) to identify and disambiguate users.

Publishing Applications to Users

This chapter describes how you use organizational hierarchies to manage Oracle Secure Global Desktop (SGD) users and give them access to applications.

This chapter includes the following topics:

- “Organizations and Objects” on page 107
- “Publishing Applications” on page 124

Organizations and Objects

SGD is built on the principles of directory services. Users, applications, and application servers are represented by *objects* in a directory. The objects are arranged into an *organizational hierarchy* representing your organization.

An organizational hierarchy starts with a top-level directory object, usually an organization object. Other directory objects, such as an organizational unit (OU), are containers that can be used to divide the organizational hierarchy. You can create group objects. Group objects are not containers. Groups have members that are objects located in other parts of the organizational hierarchy.

SGD also includes a number of different object types for representing users, applications, and application servers.

Each object has a number of configuration settings, known as *attributes*. For example, an application object has an Icon attribute that is the name of an icon to display to users.

SGD objects, and the attributes used for each object, are based on the commonly-used LDAP version 3 schema. These objects have been extended, using the standard method of doing so, to support SGD functionality. For more information on the LDAP schema, see RFC 2256 (<http://www.faqs.org/rfcs/rfc2256.html>).

SGD uses a *local repository* to store all the objects in your organizational hierarchy. Each object is distinguished from other objects in the same container by using an attribute name as a prefix, for example `ou=Sales`. This attribute is known as the *naming attribute* or the *relative distinguished name* (RDN). Two objects in the same container cannot have the same RDN. The complete name of the object that includes all the RDNs from the top of the hierarchy is the *distinguished name* (DN), for example `o=example/ou=Sales`. The DN is the name that uniquely identifies an object.

SGD object names are written like file system paths (slash-separated and top-down). The following table shows some example objects, their RDN, and their DN.

| Object Type | Relative Distinguished Name | Distinguished Name |
|--------------|--------------------------------|---|
| Organization | <code>o=example</code> | <code>o=example</code> |
| OU | <code>ou=Sales</code> | <code>o=example/ou=Sales</code> |
| User profile | <code>cn=Violet Carson</code> | <code>o=example/ou=Sales/cn=Violet Carson</code> |
| User profile | <code>cn=Elizabeth Blue</code> | <code>o=example/ou=Sales/cn=Elizabeth Blue</code> |

The relationships between objects are significant. For example, to deploy an application to users, you associate user profile objects with an application object. SGD calls these relationships *assignments*. Assignments are described in more detail in [“Publishing Applications” on page 124](#).

For more information about hierarchies and objects, see the following sections:

- [“Organizational Hierarchies” on page 108](#)
- [“SGD Object Types” on page 110](#)
- [“Designing the Organizational Hierarchy” on page 115](#)
- [“Naming Objects in the Organizational Hierarchy” on page 116](#)
- [“Populating the SGD Organizational Hierarchy Using a Batch Script” on page 117](#)
- [“LDAP Mirroring” on page 118](#)
- [“SGD Administrators” on page 122](#)

Organizational Hierarchies

SGD uses four organizational hierarchies: one each for users, applications, and application servers, and a System Objects hierarchy that contains objects for use by SGD. In the Administration Console, you use the following tabs to manage these organizational hierarchies:

- [User Profiles Tab](#)
- [Applications Tab](#)

■ Application Servers Tab

The following sections describe these tabs, the objects that they can contain, and how they are used. The System Objects organization is also described.

On the command line, you manage your organizational hierarchies with the `tarantella object` command. You can also use this command to populate an organizational hierarchy using a batch script. See “[Populating the SGD Organizational Hierarchy Using a Batch Script](#)” on page 117.

User Profiles Tab

In the Administration Console, the User Profiles tab is where you create and configure objects for managing SGD users. You use the objects on this tab to control users’ SGD-related settings, and the applications that they can access through SGD.

By default, this tab contains two objects, an organization object called `o=organization` and a domain component object called `dc=com`. These are the top-level objects in the organizational hierarchy. You can rename or delete these objects, or create new top-level objects. You create all the objects you need for managing users within these top-level objects.

The following are the SGD object types that are available on the User Profiles tab:

- [Directory Object: Organization](#)
- [Directory \(Light\) Object: Domain Component](#)
- [Directory Object: Organizational Unit](#)
- [Directory \(Light\) Object: Active Directory Container](#)
- [User Profile Object](#)

Applications Tab

In the Administration Console, the Applications tab is where you create and configure objects that represent the applications and documents that users can access through SGD. These objects are always created within the applications organization. On the command line, this organization is called `o=applications`.

The following are the SGD object types that are available on the Applications tab:

- [Directory Object: Organizational Unit](#)
- [Group Object](#)
- [X Application Object](#)
- [Windows Application Object](#)
- [Character Application Object](#)

- [Document Object](#)
- [3270 Application Object](#)
- [5250 Application Object](#)
- [Dynamic Application Object](#)

Application Servers Tab

In the Administration Console, the Application Servers tab is where you create and configure objects for managing the application servers that run the applications displayed through SGD. These objects are always created in the application servers organization. On the command line, this organization is called `o=appservers`.

The following are the SGD object types that are available on the Application Servers tab:

- [Directory Object: Organizational Unit](#)
- [Group Object](#)
- [Application Server Object](#)
- [Dynamic Application Server Object](#)

The System Objects Organization

The System Objects organization contains objects that are essential for the running and maintenance of SGD. On the command line, the System Objects organization is displayed as `o=Tarantella System Objects`.

The System Objects organization contains the Global Administrators role object. This object determines who is an SGD Administrator, and who can use the SGD graphical administration tools. See [“SGD Administrators” on page 122](#).

The System Objects organization also contains profile objects. These are default user profile objects for use with the various SGD authentication mechanisms. For example, the profile object `System Objects/LDAP Profile` is the default user profile if you are using LDAP or Active Directory authentication.

You can edit objects in the System Objects organization, but you cannot create, move, rename, or delete objects.

SGD Object Types

This section describes the available SGD object types and how they are used.

The following are the object types that are used to organize users, applications, and application servers:

- “Directory Object: Organization” on page 111
- “Directory (Light) Object: Domain Component” on page 111
- “Directory Object: Organizational Unit” on page 112
- “Directory (Light) Object: Active Directory Container” on page 112

The following are the object types used to represent users, applications, and application servers.

- “Group Object” on page 113
- “User Profile Object” on page 112
- “Windows Application Object” on page 113
- “X Application Object” on page 113
- “Character Application Object” on page 114
- “Document Object” on page 114
- “3270 Application Object” on page 114
- “5250 Application Object” on page 114
- “Dynamic Application Object” on page 115
- “Application Server Object” on page 115
- “Dynamic Application Server Object” on page 115

Directory Object: Organization

Directory objects that are organization objects are used for the things that apply to your organization as a whole. Organization objects are always at the top of the organizational hierarchy and can contain OU, Active Directory container, or user profile objects.

On the command line, you create an organization object with the `tarantella object new_org` command.

Organization objects have an `o=` naming attribute.

Directory (Light) Object: Domain Component

Directory (light) objects that are domain component objects are used to replicate a directory structure, usually a Microsoft Active Directory structure, within the SGD organizational hierarchy. Domain component objects are similar to organization objects, but do not include additional SGD-specific attributes or allow you to assign applications. This is why they are called directory (light) objects.

Domain component objects can only appear at the top of the organizational hierarchy, or within another domain component object. Domain component objects can contain OU, domain component, Active Directory container, or user profile objects.

On the command line, you create a domain component object with the `tarantella object new_dc` command.

Domain component objects have a `dc=` naming attribute.

Directory Object: Organizational Unit

Directory objects that are OU objects are used to divide your users, applications, and application servers into different departments, sites, or teams.

An OU can be contained in an organization or a domain component object.

On the command line, you create a directory object with the `tarantella object new_orgunit` command.

Directory objects have an `ou=` naming attribute.

Directory (Light) Object: Active Directory Container

Active Directory container objects are used to replicate your Microsoft Active Directory structure within the SGD organizational hierarchy.

Active Directory container objects are similar to OUs, but do not include additional SGD-specific attributes or allow you to assign applications. This is why they are called directory (light) objects.

An Active Directory container object can be contained in an organization, an OU, or a domain component object.

On the command line, you create an Active Directory container object with the `tarantella object new_container` command.

Active Directory container objects have a `cn=` naming attribute.

User Profile Object

User profile objects are used to represent a user in your organization, and give that user access to applications. They also define the SGD settings associated with a user.

How SGD associates a user profile object with a user depends on the authentication mechanisms in use. For some authentication mechanisms, you might not have to create user profile objects at all. See [“Secure Global Desktop Authentication” on page 47](#) for details.

On the command line, you create a user profile object with the `tarantella object new_person` command.

User profile objects can have a `cn=` (common name), a `uid=` (user identification), or a `mail=` (mail address) naming attribute.

Group Object

Group objects are used to associate groups of applications with an object on the User Profiles tab or groups of application servers with an object on the Applications tab.

Group objects are not the same as directory objects. Applications or application servers can only belong to one directory, but can be a member of many different groups.

Members of a group can be applications, application servers, or other groups. Groups can be moved or renamed without affecting group membership.

Groups of application server objects can be used to associate similar application servers for load balancing. See [“Load Balancing” on page 352](#) for details.

On the command line, you create a group object with the `tarantella object new_group` command.

Group objects have a `cn=` naming attribute.

Windows Application Object

Windows application objects are used to give Microsoft Windows graphical applications to users. See [“Windows Applications” on page 137](#) for more details.

On the command line, you create a Windows application object with the `tarantella object new_windowsapp` command.

Windows application objects have a `cn=` naming attribute.

X Application Object

X application objects are used to give X11 graphical applications to users. See [“X Applications” on page 153](#) for more details.

On the command line, you create an X application object with the `tarantella object new_xapp` command.

X application objects have a `cn=` naming attribute.

Character Application Object

Character application objects are used to give VT420, Wyse 60, or SCO Console character applications to users. See [“Character Applications” on page 159](#) for more details.

On the command line, you create a character application object with the `tarantella object new_charapp` command.

Character application objects have a `cn=` naming attribute.

Document Object

Document objects are used to give documents to users. A document object can refer to any Uniform Resource Locator (URL).

On the command line, you create a document object with the `tarantella object new_doc` command.

Document objects have a `cn=` naming attribute.

3270 Application Object

3270 application objects are used to give 3270 (mainframe) applications to users.

On the command line, you create a 3270 application object with the `tarantella object new_3270app` command.

3270 application objects have a `cn=` naming attribute.

5250 Application Object

5250 application objects are used to give 5250 (AS/400) applications to users.

On the command line, you create a 5250 application object with the `tarantella object new_5250app` command.

5250 Application objects have a `cn=` naming attribute.

Dynamic Application Object

Dynamic application objects are used with dynamic launch to enable users to select an application to run. See [“Dynamic Launch” on page 169](#) for details.

On the command line, you create a dynamic application object with the `tarantella object new_dynamicapp` command. Dynamic application objects have a `cn=` naming attribute.

Application Server Object

Application server objects are used to represent an application server that is used to run applications through SGD.

Application servers are used with load balancing. If you assign two or more application server objects to an application object, SGD chooses which application server to use, based on the load across the application servers. See [“Load Balancing” on page 352](#) for details.

On the command line, you create an application server object with the `tarantella object new_host` command. Application server objects have a `cn=` naming attribute.

Dynamic Application Server Object

Dynamic application server objects are used with dynamic launch to enable users to select the application server that runs the application. See [“Dynamic Launch” on page 169](#) for details.

On the command line, you create a dynamic application server object with the `tarantella object new_host --dynamic` command. Dynamic application server objects have a `cn=` naming attribute.

Designing the Organizational Hierarchy

You have complete control over the objects that you create to model your organizational hierarchy. However it is important to design and test your organizational hierarchy before implementing it. The following factors affect your design:

- **Authentication mechanism.** The most important influence on the design of organizational hierarchy is the Secure Global Desktop authentication mechanisms you use. For example, if you use UNIX system authentication, you can structure

the hierarchy however you like. However, with LDAP authentication, you might need to mirror part of your LDAP directory structure. See [“Secure Global Desktop Authentication” on page 47](#) for details.

- **Organization chart.** Sometimes it is a good approach to use OUs to represent the departments or offices in your organization. However, if your organization is restructured, you might have to reorganize your hierarchy.
- **Inheritance.** The settings for user profile objects and OU objects can be inherited from the object’s parent in the organizational hierarchy. For example if everyone in a department needs an application, assign the application to the OU that represents the department. Every user belonging to that OU gets the applications assigned to the OU. Inheritance works best if you are *not* using LDAP assignments.
- **User profile objects.** User profile objects can be configured to give users access to particular applications and customized settings. Depending on the authentication mechanisms you enable, a default user profile is often used and this might be sufficient for your needs. This is particularly true if you use LDAP assignments to assign applications to users.
- **Naming convention.** Use a naming convention for each application or document object type. The name of the application or document object is displayed to users. For user profile objects, it is best to use the person’s full name, for example “Indigo Jones”.

Naming Objects in the Organizational Hierarchy

When you create an object in the Administration Console, you can use any characters you want for the name of the object, apart from backslash (\) or plus (+).

On the command line, if you use a forward slash in an object name, you must backslash protect, or escape, it. This is because SGD interprets the forward slash as a part of the organizational hierarchy. For example, if you try to create an object with the relative name `cn=a/b` beneath `o=organization`, SGD tries to create an object called `b` within `o=organization/cn=a`. This fails because `o=organization/cn=a` does not exist. To create an object with this name, type `cn=a\b`.

On the command line, if the name of an object includes spaces, make sure you enclose the name in quotes, for example `".../_ens/o=Indigo Insurance"`.

With the `tarantella object` command, any name in the local repository is treated as case insensitive. When you create or rename an object, the case used is preserved. However, other commands, such as the `tarantella webtopsession` and `tarantella emulatorsession` commands, *are* case sensitive.

Populating the SGD Organizational Hierarchy Using a Batch Script

If you want to populate your organizational hierarchy with a large number of objects, using the Administration Console to do this is not very efficient. The solution is to use the batch scripting functionality of the `tarantella object` command.

Once you have designed the structure of your SGD organizational hierarchy, you create a file for each type of object you want. Each file contains one line per object, with the correct syntax for creating the object from the appropriate `tarantella object` command. For example, to create five OUs you might have a file called `orgunits.txt` containing the following:

```
--name "o=Indigo Insurance/ou=IT" \  
--name "o=Indigo Insurance/ou=Sales" \  
--name "o=Indigo Insurance/ou=Marketing" \  
--name "o=Indigo Insurance/ou=Finance" \  
--name "o=Indigo Insurance/ou=Finance/ou=Administration"
```

Do not include the actual `tarantella object` command name, for example `object new_orgunit`, as part of each line.

Remember the following:

- Application objects, including their groups and OUs, must be created in the `o=applications` organization.
- Application server objects, including their groups and OUs, must be created in the `o=appservers` organization.
- Every application must have an application object.
- Every application server must have an application server object.

Once all your files are complete, use the `tarantella object script` command to process them all at once, for example:

```
#!/bin/sh  
tarantella object script << EOF  
new_orgunit --file orgunits.txt  
new_group --file groups.txt  
new_host --file hosts.txt  
new_person --file people.txt  
new_xapp --file xapps.txt  
new_windowsapp --file windowsapps.txt  
new_charapp --file charapps.txt  
EOF
```

The `tarantella object script` command runs each command in order. Each command reads and processes the specified file.

You can use any `tarantella object` subcommand with the `tarantella object script` command. You do not have to read in object details from other files.

Many other commands, for example the `tarantella passcache` command, accept `--file` arguments so you can perform multiple related actions at once.

LDAP Mirroring

When a user is authenticated with either LDAP authentication, Active Directory authentication, or third-party authentication using the LDAP search, SGD establishes the user profile for a user by searching the local repository, allowing for differences between the LDAP and SGD naming systems. SGD searches for the following until a match is found:

- A user profile with the same name as the user's LDAP object.
For example, if the LDAP object is `cn=Emma Rald,cn=Sales,dc=example,dc=com`, SGD searches the local repository for `dc=com/dc=example/cn=Sales/cn=Emma Rald`.
- A user profile in the same organizational unit as the LDAP object but with the name `cn=LDAP Profile`.
For example, `dc=com/dc=example/cn=Sales/cn=LDAP Profile`.
- A user profile in any parent organizational unit with the name `cn=LDAP Profile`.
For example, `dc=com/dc=example/cn=LDAP Profile`.

If there is no match, the profile object `o=System Objects/cn=LDAP Profile` is used for the user profile.

Typically LDAP and Active Directory users use the default LDAP profile, and applications and documents are assigned to them using LDAP assignments. See [“LDAP Assignments” on page 127](#). However, user profile objects can also be used to control a user's SGD-specific settings, such as the ability to use copy and paste or to edit client profiles. If you want to customize an LDAP or Active Directory user's SGD settings, you might have to mirror some of your LDAP structure in the local repository.

When you mirror your LDAP structure, remember the following:

- Do not mirror your entire LDAP structure in the local repository. Only create as much of the structure as you need.
- Inherit as much as possible from other objects in the organizational hierarchy.

- Do not create user profile objects for all users. Only create user profile objects for users that must have individual settings. Creating `cn=LDAP Profile` objects is sufficient in most cases.

You can configure service objects that specify a base DN (a search root) as part of the LDAP URL, see [“Using Service Objects” on page 90](#). The base DN can be used as the starting point when mirroring your LDAP structure. SGD only permits an organization object (`o=`) or domain component (`dc=`) object as the top-level object. If your LDAP structure uses other objects, such as country (`c=`) or location (`l=`) objects, you must ensure the base DN for the service object enables you to mirror from an organization or domain component object. SGD also constrains the objects you can use as directory containers. For example, you cannot nest an organization object inside an organization object. This means you might have to create service objects with different base DN in order to mirror everything you need.

When working with LDAP mirroring in the Administration Console, it is useful to display the naming attribute for the objects you work with. By default the Administration Console does not display naming attributes. You enable the display of naming attributes in the Preferences for the Administration Console.

When working with user profiles in the Administration Console, select Local + LDAP from the Repository list on the User Profiles tab. LDAP objects that are mirrored in the local repository are indicated by the following icon:



The following is an example of how to mirror your LDAP organization to give users different SGD settings.

An Example of LDAP Mirroring

Indigo Insurance has five departments: IT, Sales, Marketing, Finance, and Administration. The Finance and Marketing departments need different SGD settings to the other departments. Sid Cerise in the Finance department needs different SGD settings to the other users in the Finance department.

The objects you create depend on the type of LDAP directory server used, as described in the following sections.

Oracle Directory Server Enterprise Edition

For Oracle Directory Server Enterprise Edition (formerly Sun Java System Directory Server), the following are the LDAP names of the objects you need to mirror in the local repository and the object types use:

- `o=indigo-insurance.com`

Use an organization object.

- `ou=Finance,o=indigo-insurance.com`

Use an OU object.

- `ou=Marketing,o=indigo-insurance.com`

Use an OU object.

Note – In the Administration Console, create Directory objects. The naming attribute is set automatically.

FIGURE 3-1 shows the mirrored objects in the Administration Console.

FIGURE 3-1 Example Mirrored LDAP Objects for Oracle Directory Server



With this structure in place, create the following user profile objects in the local repository:

- `o=indigo-insurance.com/ou=Finance/cn=LDAP Profile`
- `o=indigo-insurance.com/ou=Marketing/cn=LDAP Profile`
- `o=indigo-insurance.com/ou=Finance/uid=Sid Cerise`

Note – In the Administration Console, remember to select uid as the naming attribute for the user profile object `o=indigo-insurance.com/ou=Finance/uid=Sid Cerise`.

With this organizational hierarchy, users receive settings as follows:

- Sid Cerise receives the settings defined for the following user profile object, including any settings inherited from parent objects in the organizational hierarchy:
`o=indigo-insurance.com/ou=Finance/uid=Sid Cerise`
- Users in the Finance department receive the settings defined for the following user profile object, including any settings inherited from parent objects in the organizational hierarchy:
`o=indigo-insurance.com/ou=Finance/cn=LDAP Profile`

- Users in the Marketing department receive the settings defined for the following user profile object, including any settings inherited from parent objects in the organizational hierarchy:

`o=indigo-insurance.com/ou=Marketing/cn=LDAP Profile`

- All other users receive the settings defined for the default LDAP user profile, System Objects/cn=LDAP Profile

Microsoft Active Directory

For Microsoft Active Directory, the following are the LDAP names of the objects you need to mirror in the local repository and the object types to use:

- `dc=indigo-insurance,dc=com`
Use a domain component object.
- `cn=Finance,dc=indigo-insurance,dc=com`
Use an Active Directory container object.
- `cn=Marketing,dc=indigo-insurance,dc=com`
Use an Active Directory container object.

Note – In the Administration Console, you create domain components and Active Directory containers by creating Directory (light) objects, and then selecting the correct naming attribute.

FIGURE 3-2 shows the mirrored objects in the Administration Console.

FIGURE 3-2 Example Mirrored LDAP Objects for Microsoft Active Directory



With this structure in place, create the following user profile objects in the local repository:

- `dc=com/dc=indigo-insurance/cn=Finance/cn=LDAP Profile`
- `dc=com/dc=indigo-insurance/cn=Marketing/cn=LDAP Profile`
- `dc=com/dc=indigo-insurance/cn=Finance/cn=Sid Cerise`

With this organizational hierarchy, users receive settings as follows:

- Sid Cerise receives the settings defined for the following user profile object:
`o=indigo-insurance.com/cn=Finance/cn=Sid Cerise`
- Users in the Finance department receive the settings defined for the following user profile object:
`o=indigo-insurance.com/ou=Finance/cn=LDAP Profile.`
- Users in the Marketing department receive the settings defined for the following user profile object:
`o=indigo-insurance.com/ou=Marketing/cn=LDAP Profile.`
- All other users receive the settings defined for the default LDAP user profile, `System Objects/cn=LDAP Profile`

Note – It is not possible to inherit SGD settings from domain component and Active Directory container objects.

SGD Administrators

In SGD, administration privileges are managed using the Global Administrators role object in the System Objects organization.

The Global Administrators role object has a list of members, and a list of assigned applications. All SGD Administrators are defined as members of the Global Administrators role object. The list of assigned applications is used to assign administration tools to SGD Administrators. SGD Administrators are assigned these applications *in addition to* any other applications assigned to them.

Only SGD Administrators can configure SGD using the SGD graphical administration tools, Administration Console and Profile Editor. To use the SGD command-line tools, the following conditions apply:

- Commands that control the SGD server and SGD web server can be run only by superuser (root).
- Commands for creating and managing arrays of SGD servers can only be run by SGD Administrators.
- All other commands can be run by any user in the `ttaserv` group.

Use the `usermod -G` command to make a user a member of the `ttaserv` group. The `ttaserv` group does not have to be the users primary or effective group.

You can use the SGD Administration Console or the `tarantella role` command to add or remove SGD Administrators.

If no user profile objects are defined as members of the Global Administrators role object, the UNIX or Linux system root user has administration privileges.

Note – If you want SGD Administrators to authenticate using an LDAP directory or Active Directory authentication, you must create user profiles for them. See [“LDAP Mirroring” on page 118](#) for details.

▼ How To Add an SGD Administrator

1. In the Administration Console, go to the User Profiles tab.
2. Select the Global Administrators role object.
 - a. In the navigation tree, click System Objects.
The System Objects table is displayed.
 - b. In the System Objects table, click the Global Administrators role object.
The Members tab is displayed.
3. Add a user profile object to the Members tab.
 - a. In the Editable Members table, click Add.
The Add User Assignment window is displayed.
 - b. Locate the user profile object.
Use the Search field or the navigation tree to find the object you want.
 - c. Select the check box next to a user profile object.
To add several SGD Administrators, select more than one user profile object.
 - d. Click Add Assignment.
The Members tab is displayed, showing the selected user profile object.

Tip – You can also use the `tarantella role add_member --role global -member pobj` command.

▼ How To Remove an SGD Administrator

1. In the Administration Console, go to the User Profiles tab.
2. Select the Global Administrators role object.

- a. In the navigation tree, click **System Objects**.

The System Objects table is displayed.

- b. In the System Objects table, click the **Global Administrators** role object.

The Members tab is displayed.

3. Remove a user profile object from the Members tab.

- a. In the Editable Members table, select the check box next to a user profile object.

To remove several SGD Administrators, select more than one user profile object.

- b. Click **Delete**.

A warning message is displayed.

- c. Click **OK**.

The Members tab is displayed.

Tip – You can also use the `tarantella role remove_member --role global --member pobj` command.

Publishing Applications

Creating objects to represent the applications, application servers, and users in your organization does not, by itself, give users to access applications through SGD. Applications must be published. You publish applications by creating relationships between the objects in the organizational hierarchy. SGD calls these relationships *assignments*. You publish applications as follows:

- **Assign applications to application servers.** This configures the application servers that can run the application.
- **Assign applications to users.** This configures the users that see the application on their webtop.

Assignments can be either of the following types:

- **Local assignments.** These are relationships between objects that are in the SGD repository. See [“Local Assignments” on page 125](#).
- **LDAP assignments.** These are relationships between objects in the SGD repository and objects in an LDAP directory. See [“LDAP Assignments” on page 127](#).

Assigning applications to application servers is done by using local assignments.

Assigning applications to users is done by using local assignments, LDAP assignments, or a combination of both.

The Administration Console provides several ways for reviewing assignments, see [“Reviewing Assignments” on page 132](#).

Local Assignments

Local assignments are relationships between objects in the local repository.

In the Administration Console, you assign applications on the Applications tab as follows:

- Use the Hosting Application Servers tab to assign applications, or groups of applications, to application servers.

See [“How to Assign Application Servers to Applications” on page 126](#).

Tip – You can also assign applications from the Hosted Applications tab for group and application server objects.

- Use the Assigned User Profiles tab to assign applications to users.

See [“How to Assign Applications to Users” on page 126](#).

Tip – You can also assign applications from the Assigned Applications tab for directory and user profile objects.

SGD uses inheritance to make local assignments easier to manage and more efficient. OU and user profile objects can inherit the assignments and settings of their parent objects in the organizational hierarchy. Inheritance is enabled by default. To use inheritance, create user profile objects within OU objects, and then assign applications to the OUs.

The Administration Console provides several ways for reviewing assignments, see [“Reviewing Assignments” on page 132](#).

▼ How to Assign Application Servers to Applications

1. **In the Administration Console, go to the Applications tab and select an application object or a group object.**

If you select a group of applications, you can assign application servers to all the applications in the group.

The General tab is displayed.

2. **Go to the Hosting Application Servers tab.**

3. **In the Editable Assignments table, click Add.**

The Add Application Server Assignment window is displayed.

4. **Locate application server or group objects.**

Use the Search field or the navigation tree to find the objects you want.

5. **Select the check box next to the application server or group objects and click Add**

If you select more than one application server, or a group of application servers, SGD load balances between application servers. See [“Load Balancing” on page 352](#).

If you select a group of application servers, you select all the application servers in the group.

The Effective Application Servers table is updated with the selected application servers.

▼ How to Assign Applications to Users

1. **In the Administration Console, go to the Applications tab and select an application object or a group object.**

If you select a group of applications, you can assign all the applications in the group to users.

The General tab is displayed.

2. **Click the Assigned User Profiles Tab.**

3. **In the Editable Assignments table, click Add.**

The Add User Assignment window is displayed.

4. Locate user profile or directory objects.

Use the Search field or the navigation tree to find the objects you want.

You can assign an application to user profile or directory objects.

If you assign an application to a directory object, all the user profiles contained in that directory object automatically receive the application. This is called *inheritance*. Assigning an application to directory objects is more efficient.

5. Select the check box next to the user profile or directory objects and click Add.

The Effective User Profiles table is updated with the selected users.

LDAP Assignments

LDAP assignments make use of SGD's Directory Services Integration feature. With Directory Services Integration, you use an LDAP directory instead of the local repository for holding user information. This means you do not need to create user profile objects in the local repository.

You can only use Directory Services Integration for users who have their user identity established by searching an LDAP directory or Active Directory. This means users must be authenticated by one of the following authentication mechanisms:

- Active Directory authentication, see [“Active Directory Authentication” on page 52](#)
- LDAP authentication, see [“LDAP Authentication” on page 63](#)
- Third-party authentication using the LDAP repository search, see [“Third-Party Authentication” on page 72](#)

LDAP assignments are relationships between objects in the SGD repository and objects in an LDAP directory. With LDAP assignments, instead of assigning applications to users, you assign users to applications. In the Administration Console, you do this on the Assigned User Profiles tab for application, document, and group objects. You can assign users as follows:

- **LDAP users.** You select individual users in an LDAP directory.

See [“How to Assign Applications to LDAP Users” on page 128](#) for details.

- **LDAP groups.** You select groups in an LDAP directory and SGD assigns the users in the group to the application.

See [“How to Assign Applications to Members of LDAP Groups” on page 129](#) for details.

You might have to perform additional configuration to use LDAP group searches successfully. See [“Tuning LDAP Group Searches” on page 132](#) for details.

- **LDAP searches.** You configure an LDAP search filter or URL and SGD assigns the matching users to the application.

See [“How to Assign Applications Using LDAP Searches” on page 130](#) for details.

When working with LDAP assignments in the Administration Console, it is useful to display the naming attribute for the objects you work with. By default the Administration Console does not display naming attributes. You enable the display of naming attributes in the Preferences for the Administration Console.

If you want more control over the SGD-specific settings for LDAP users, such as the ability to use copy and paste, or to edit client profiles, see [“LDAP Mirroring” on page 118](#).

The Administration Console shows you which users are configured to receive an application using LDAP assignments, see [“Reviewing Assignments” on page 132](#).

SGD caches the directory data it obtains, see [“Managing the Directory Services Cache” on page 134](#) for more details.

See [“Troubleshooting LDAP Assignments” on page 135](#) for tips on working with LDAP assignments.

▼ How to Assign Applications to LDAP Users

1. In the SGD Administration Console, go to the Applications tab.

2. Select an application or group object and go the Assigned User Profiles tab.

Use the Search field or the navigation tree to find the object you want.

If you select a group object, LDAP users receive all the applications in the group.

3. In the Editable Assignments table, click the Add button.

The Add User Assignment window is displayed.

4. From the Repository list, select Local + LDAP.

5. (Optional) Select a service object from the View list.

By default, the first enabled service object in the list of service objects is selected. Only enabled service objects are available in the View list. See [“Using Service Objects” on page 90](#).

6. Locate the LDAP users you want to assign to the object.

Use the Search field or the navigation tree to find users in the LDAP directory.

7. Select the check box next to the LDAP users and click the Add button.

If you assign several LDAP users to an object, it is more efficient to use an LDAP search.

Tip – On the command line, you can use the `--ldapusers` option to assign LDAP users.

The Add User Assignment window closes and the Editable Assignments table is updated with the LDAP users.

▼ How to Assign Applications to Members of LDAP Groups

1. In the Administration Console, go to the Applications tab.

2. Select an application, document, or group object and go to the Assigned User Profiles tab.

Use the Search field or the navigation tree to find the object you want.

If you select a group object, all members of the LDAP group receive all the applications in the group.

3. In the Editable Assignments table, click the Add button.

The Add User Assignment window is displayed.

4. From the Repository list, select Local + LDAP.

5. (Optional) Select a service object from the View list.

By default, the first enabled service object in the list of service objects is selected. Only enabled service objects are available in the View list. See [“Using Service Objects” on page 90](#).

6. Locate the LDAP groups you want to assign to the object.

Use the Search field or the navigation tree to find groups in the LDAP directory.

7. Select the check box next to the LDAP groups and click the Add button.

If you assign several groups to an object, it is more efficient to use LDAP searches instead.

Tip – On the command line, you can use the `--ldapgroups` option to assign the members of LDAP groups.

The Add User Assignment window closes and the Editable Assignments table is updated with the LDAP groups.

▼ How to Assign Applications Using LDAP Searches

1. In the Administration Console, go to the Applications tab.
2. Select an application, document, or group object and go to the Assigned User Profiles tab.
3. In the LDAP Searches section configure the LDAP search.

Do either of the following:

- Select the Simple Search option and use the LDAP query builder to construct the LDAP search.
- Select the Advanced Search option and enter the LDAP search string in the LDAP URL or Filter field.

See “Using LDAP Searches” on page 130 for details.

Use the Preview button to check whether the configured search returns the expected results.

Tip – On the command line, you can use the `--ldapsearch` option to configure LDAP searches.

4. Click Save.

Using LDAP Searches

LDAP searches can be either of the following:

- An RFC 2254 search filter, see (<http://www.faqs.org/rfcs/rfc2254.html>)
- An RFC 1959 LDAP URL, see (<http://www.faqs.org/rfcs/rfc1959.html>)

The Administration Console provides a Simple Search and an Advanced Search for configuring LDAP searches.

Note – The Administration Console does not automatically escape the special characters specified in RFC2254. To use a special character in the Administration Console, you must manually type the escape sequence. For example, to search for a user with the common name “John Doe (123456)”, type the following `cn=John Doe\0x28123456\0x29` in the search field.

SGD supports the use of extensible matching search filters as specified in [RFC2254](#). This enables you to look up information from components that make up an object's DN. For example, to assign an application to a user that is contained within any OU called managers (ou=managers), you can use a (&(ou:dn:=managers)) search filter. Active Directory does not support extensible search filters.

As you configure LDAP searches, use the Preview button to check that the search returns the expected results.

Using the Simple Search

The Simple Search enables you to construct an LDAP search using the following commonly-used LDAP and Active Directory attributes.

| Attribute Name | Description |
|------------------|--|
| c | The countryName attribute containing a two-letter ISO 3166 country code. |
| cn | The commonName attribute containing the name of the object. For person objects, this is usually the person's full name. |
| departmentNumber | The attribute containing the code for a department. The code can be numeric or alphanumeric. |
| l | The localityName attribute containing the name of a locality such as a city or country. |
| memberOf | The commonly-used attribute for managing users in Active Directory. Contains a list of groups to which the user belongs. |
| sn | The surname attribute containing the family name of a person. |

Click the Browse button to display the Select Root for LDAP Search window. This window enables you to select an LDAP object to use as the search root. If you have configured more than one service object, use the View list to select a service object to use for the search root. Only enabled service objects are available in the View list. If you specify a search root, the search is formatted as an LDAP URL. If you do not specify a search root, the search is formatted as an LDAP filter. The filter is applied to all the enabled service objects.

When you save a Simple Search, the search string is displayed in the Advanced Search field.

Using the Advanced Search

The Advanced Search field enables you to enter your own LDAP search filter or URL, or to paste in a search from another tool.

If you enter an LDAP URL, use the format `ldap://search`. If you include the host, port, and return attribute specification in the URL they are ignored.

You can use the Simple Search to construct a basic search and save it. This loads the simple search into the Advanced Search field. Then select the Advanced Search option to fine tune the search.

Note – If you fine tune a Simple Search in the Advanced Search field and edit it in a way that is not compatible with a Simple Search, you might not be able to edit the search again as a Simple Search. If this happens, you must clear the Advanced Search field and save the change. Then rebuild the Simple Search.

Reviewing Assignments

The Administration Console enables you to review assignments as follows:

- Assigned User Profiles tab for application, document, group, and OU objects – The Effective User Profiles table shows you the users that are assigned the application
- Assigned Applications tab for user profile, OU, and organization objects – The Effective Applications table shows you the applications that are assigned to users
- Hosting Application Servers tab on application and group objects – The Effective Application Servers table shows you the application servers that can run an application
- The Hosted Applications tab on application server and group objects – The Effective Applications table shows you the applications that can run on the application servers
- The Members tab on group objects – The Effective Members table shows you the members of the group

By default, LDAP assignments are not displayed. To display LDAP assignments, click the Load LDAP link in the effective assignment tables.

The effective assignment tables enable you to trace the origin of assignments, where the assignment is the result of inheritance, group membership, or an LDAP search.

Tuning LDAP Group Searches

You can tune the LDAP group searches to return the users you require for LDAP assignments by configuring how SGD identifies the users in a group and whether SGD can search nested groups or sub-groups.

Increasing the Group Search Depth

By default, the LDAP group search does not search nested groups or sub-groups. If your organization uses nested groups or sub-groups, you can increase the depth of the search. Increasing the depth might have a negative effect on performance.

To increase the depth of group searches, use the following command:

```
$ tarantella config edit \  
--tarantella-config-ldap-nested-group-depth depth
```

The default *depth* is 0. Increase the value of *depth* to match the depth of the nested groups.

Group Membership Attributes

SGD establishes group membership by searching for attributes on LDAP user objects and LDAP group objects. LDAP user objects are checked before LDAP group objects.

User group membership attributes are attributes on LDAP user objects that list the groups to which the users belong. By default, SGD searches for groups in the `isMemberOf`, `nsroledn`, `memberOf` attributes on LDAP user objects. To configure the user group membership attributes, use the following command:

```
$ tarantella config edit \  
--tarantella-config-ldap-object-member-attributes attribute ...
```

You can list more than one *attribute*. Each *attribute* must be separated by a space. Remember to include the default attributes `isMemberOf`, `nsroledn`, `memberOf` in the list.

Group user membership attributes are attributes on LDAP group objects that list the users that belong to the group. By default, SGD searches for users in the `uniquemember` and `member` attributes on LDAP group objects. To configure the group user membership attributes, use the following command:

```
$ tarantella config edit \  
--tarantella-config-ldap-group-member-attributes attribute ...
```

You can list more than one *attribute*. Each *attribute* must be separated by a space. Remember to include the default attributes `uniquemember` and `member` in the list.

Short Attributes

If the group membership attributes do not contain the DN's of users, then the group search fails.

You can configure SGD to search short attributes that can be used to identify users. For short attributes to work, they must contain unique values. Short attributes can be on LDAP user objects or LDAP group objects.

To configure SGD to search short attributes on LDAP user objects, use the following command:

```
$ tarantella config edit \  
--tarantella-config-ldap-object-short-attributes attribute ...
```

You can list more than one *attribute*. Each *attribute* must be separated by a space.

To configure SGD to search short attributes on LDAP group objects, use the following command

```
# tarantella config edit \  
--tarantella-config-ldap-group-short-attributes attribute ...
```

You can list more than one *attribute*. Each *attribute* must be separated by a space.

Managing the Directory Services Cache

SGD caches the directory services data it obtains.

If you find that SGD is not detecting changes, you can flush, refresh, or populate the cache manually with the `tarantella cache` command.

To update the cache of group data, use the following command:

```
$ tarantella cache --refresh ldapgroups
```

When you run this command, SGD searches the cache for LDAP groups, queries the directory for the membership of each LDAP group, and then adds the list of users to the cache.

To add group data to the cache, use the following command:

```
$ tarantella cache --populate ldapgroups
```

When you run this command, SGD searches the local repository for objects with LDAP group assignments and adds the LDAP groups to the cache. SGD then queries the directory for the membership of each LDAP group and adds the list of users to the cache.

To remove group data from the cache, use the following command:

```
$ tarantella cache --flush ldapgroups
```

To remove the LDAP search data from the cache, use the following command:

```
$ tarantella cache --flush ldapconn-lookups
```

To reset all LDAP connections, use the following command:

```
$ tarantella cache --flush ldapconn
```

To remove all LDAP data from the cache, use the following command:

```
$ tarantella cache --flush all
```

By default SGD keeps group data in the cache for 4300 seconds (12 hours). You might want to change how long SGD keeps group data depending on how frequently your LDAP data changes. You do this with the following command:

```
# tarantella config edit \  
--tarantella-config-ldap-ldapgroups-timeout secs
```

Troubleshooting LDAP Assignments

If LDAP group searches are not returning the expected results, see [“Tuning LDAP Group Searches” on page 132](#).

SGD caches the data it collects from an LDAP directory. If you find that SGD is not detecting changes, you can flush the cached data manually. See [“Managing the Directory Services Cache” on page 134](#).

You can configure an LDAP timeout in the event that the LDAP searches of an LDAP directory fail. See [“LDAP Operation Timeout” on page 99](#).

To help diagnose problems with LDAP assignments, set the following log filters:

```
server/webtop/*:ldapwebtop%%PID%%.log
```

```
server/webtop/*:ldapwebtop%%PID%%.jsl  
server/directoryservices/*:ldapwebtop%%PID%%.log  
server/directoryservices/*:ldapwebtop%%PID%%.jsl
```

See [“Using Log Filters to Troubleshoot Problems With an SGD Server”](#) on page 391 for more information on configuring and using log filters.

The Administration Console has some configuration settings that affect the display of LDAP data, for example the attributes that are used to identify users. If you find that LDAP operations in the Administration Console do not work as you expect, you might have to adjust the settings. See [“Administration Console Configuration Settings”](#) on page 383 for details.

Configuring Applications

This chapter contains advice on configuring applications that users can run through Oracle Secure Global Desktop (SGD), and how to diagnose and fix problems with applications.

This chapter includes the following topics:

- [“Windows Applications” on page 137](#)
- [“X Applications” on page 153](#)
- [“Character Applications” on page 159](#)
- [“Dynamic Launch” on page 169](#)
- [“Using SSH” on page 178](#)
- [“Application Authentication” on page 183](#)
- [“Tips on Configuring Applications” on page 189](#)
- [“Troubleshooting Applications” on page 202](#)

Windows Applications

This section describes how to configure Windows application objects.

This section includes the following topics:

- [“Configuring Windows Application Objects” on page 138](#)
- [“Creating Windows Application Objects on the Command Line” on page 140](#)
- [“Configuring Microsoft Windows Terminal Services for Use With SGD” on page 140](#)
- [“Licensing Microsoft Windows Terminal Services” on page 144](#)
- [“Microsoft Windows Remote Desktop” on page 145](#)

- [“Seamless Windows” on page 146](#)
- [“Key Handling for Windows Terminal Services” on page 147](#)
- [“Returning Client Device Information for Windows Terminal Services Sessions” on page 149](#)
- [“The SGD Remote Desktop Client” on page 149](#)
- [“Running Windows Applications on Client Devices” on page 152](#)

Configuring Windows Application Objects

You use a Windows application object if you want to give a Microsoft Windows graphical application to users.

In the Administration Console, the configuration settings for Windows application objects are divided into the following tabs:

- **General tab** – These settings control the name and the icon used when creating links for users
- **Launch tab** – These settings control how the application is started and whether application sessions can be suspended and resumed
- **Presentation tab** – These settings control how the application is displayed to users
- **Performance tab** – These settings are used to optimize the performance of the application
- **Client Device tab** – These settings control how the user’s client device interacts with the application

The following table lists the most commonly used settings for configuring Windows application objects, and how to use them.

| Attribute | Description |
|---------------------------|--|
| Name | The name that users see. |
| Icon | The icon that users see. |
| Application Command | The full path to the application that runs when users click the link. The application must be installed in the same location on all application servers. Leave this field blank if you want to run a Windows desktop session. |
| Arguments for Command | Any command-line arguments to use when starting the application. |
| SGD Remote Desktop Client | By default, the SGD Remote Desktop Client is used to run the application on the Microsoft Windows application server. SGD uses the Microsoft RDP protocol to connect to the application server. See “Configuring Microsoft Windows Terminal Services for Use With SGD” on page 140 . |

| Attribute | Description |
|---------------------------------|---|
| Local Client Launch | Select the Local Client Launch to run the application on the user's client device. See "Running Windows Applications on Client Devices" on page 152 . |
| Domain Name | The Windows domain to use for the application server authentication process. This can be left blank. The domain can also be configured on either the application server or the user profile. See also "Windows Domains and the Password Cache" on page 187 . |
| Number of Sessions | The number of instances of an application a user can run. The default is three. |
| Application Resumability | For how long the application is resumable. The following options are available: <ul style="list-style-type: none"> • Never – The application can never be resumed • During the User Session – The application keeps running and is resumable until the user logs out of SGD • General – The application keeps running for a time, controlled by the Timeout setting, after the user logs out of SGD, and can be resumed when the user next logs in |
| Window Type | How the application is displayed to the user. Use Kiosk for full-screen desktop sessions. Selecting the Scale to Fit Window check box for the Window Size enables SGD to scale the application window to fit the client device display. For Independent Window, you must specify a Height and Width for the Window Size or select the Client's Maximum Size check box. Use Seamless Window mode to the application in the same way it displays on the Windows application server, regardless of the user's desktop environment. See "Seamless Windows" on page 146 . |
| Color Depth | The application's color depth. See "Color Depth" on page 143 for more details. |
| Application Load Balancing | How SGD chooses the best application server to run the application. See "Application Load Balancing" on page 362 for more details. |
| Hosting Application Servers tab | Use the Editable Assignments table to select the application servers, or group of application servers, that can run the application. The application must be installed in the same location on all application servers |
| Assigned User Profiles tab | Use the Editable Assignments table to select the users that can see the application. Selecting Directory or Directory (light) objects enables you to give the application to many users at once. You can also use a Lightweight Directory Access Protocol (LDAP) directory to assign applications. See "LDAP Assignments" on page 127 . |

In addition to this configuration, you can also configure the following:

- Printing – See ["Printing" on page 223](#).
- Client drives – See ["Client Drive Mapping" on page 258](#).
- Audio – See ["Audio" on page 276](#).

- Smart cards – See “Smart Cards” on page 292.
- Copy and paste – See “Copy and Paste” on page 287.
- Serial ports – See “Serial Ports” on page 298.

Creating Windows Application Objects on the Command Line

On the command line, you create an Windows application object with the `tarantella object new_windowsapp` command. You can also create multiple Windows application objects at the same time with the `tarantella object script` command. See “Populating the SGD Organizational Hierarchy Using a Batch Script” on page 117.

Windows application objects can only be created in the `o=applications` organizational hierarchy.

Configuring Microsoft Windows Terminal Services for Use With SGD

Configuring a Windows application object enables you to use the features of Microsoft Windows Terminal Services.

Note – From Windows Server 2008 R2, Terminal Services is renamed Remote Desktop Services.

The Terminal Services features supported by SGD and the application server platforms on which they are supported are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

There are many possible configuration settings for Microsoft Windows Terminal Services. For detailed information on configuring Terminal Services, see your system documentation. To use Terminal Services with SGD, the settings you might have to configure include the following:

- “Authentication Settings” on page 141
- “Session Resumability and Session Directory” on page 141
- “Windows Printer Mapping” on page 142
- “Drive Redirection” on page 142
- “Encryption Level” on page 142

- “Multiple Terminal Services Sessions” on page 142
- “Remote Desktop Users” on page 142
- “Time Zone Redirection” on page 143
- “Audio Redirection” on page 143
- “Smart Card Device Redirection” on page 143
- “COM Port Mapping” on page 143
- “Color Depth” on page 143
- “Transport Layer Security” on page 144
- “Terminal Services Group Policies” on page 144
- “Keep Alive Configuration for Windows Terminal Servers” on page 144

Note – Changes to your Terminal Services configuration only take effect for new Windows Terminal Server sessions.

Authentication Settings

You must configure Windows Terminal Services so that it does not prompt for a password when a user logs in.

By default, Windows 2000 Server always prompts for a password when users log in, whether or not SGD supplies the password for the application server from its password cache. By default, Windows Server 2003 or later, does not prompt for passwords.

Session Resumability and Session Directory

With Windows Terminal Services, users’ sessions can continue to run following a connection loss.

If you are not using Session Directory, it is best to disable this feature on the Windows Terminal Server, and let SGD handle session resumability. This prevents unnecessary use of resources on the application server, and ensures that if users share accounts on the application server, they do not resume each other’s Windows sessions. To disable this feature, you must select End Session for the When Session Limit Is Reached Or Connection Is Broken option in Terminal Services Configuration.

If you are using Session Directory to handle session resumability, you must select Suspend Session for the When Session Limit Is Reached Or Connection Is Broken option in Terminal Services Configuration. To use Session Directory, you must also configure the Window Close Action attribute for Windows application objects to End Application Session.

Windows Printer Mapping

To support printing to client printers from a Windows Terminal Server session, Windows printer mapping must be enabled. Windows printer mapping is enabled by default.

Drive Redirection

To support mapping of client drives in a Windows Terminal Server session, drive redirection must be enabled. Drive redirection is enabled by default.

Encryption Level

You can only use the Low, Client-compatible, or High encryption levels with SGD. SGD does not support the Federal Information Processing Standards (FIPS) encryption level.

Multiple Terminal Services Sessions

By default, a Microsoft Windows Server only allows users to start one Terminal Services session. If a user starts another desktop session, or another instance of an application with the same arguments, the second Terminal Services session *grabs* the first session and disconnects it. This means that it is not possible to start two desktop sessions, or two instances of the same application, on the same Windows Server.

On Microsoft Windows Server 2003 or later application servers, you can enable support for multiple Terminal Services sessions.

Remote Desktop Users

For Microsoft Windows Server 2003 or later application servers, users can only use Terminal Services if they are members of the Remote Desktop Users group.

Time Zone Redirection

Client computers can redirect their time zone settings to the Terminal Server, so that users see the correct time for their time zone in their desktop or application sessions. Terminal Services uses the server base time on the Terminal Server and the client time zone information to calculate the time in the session. This feature is useful if you have client devices in different time zones. By default, this feature is disabled.

In the Administration Console, the Time Zone Map File attribute on the Global Settings → Client Device tab specifies a file that contains mappings between UNIX platform client device and Windows application server time zone names.

Audio Redirection

To play audio from a Windows Terminal Server session, audio redirection must be enabled on the application server. By default, audio redirection is disabled.

Smart Card Device Redirection

To use a smart card reader from a Windows Terminal Server session, smart card device redirection must be enabled on the application server. By default, smart card device redirection is enabled.

COM Port Mapping

To access the serial ports on the client device from a Windows Terminal Server session, COM port mapping must be enabled on the application server. By default, COM port mapping is disabled.

Color Depth

SGD supports 8-bit, 16-bit, 24-bit, and 32-bit color depths in a Windows Terminal Server session.

32-bit color is available on Windows Vista, Windows Server 2008, Windows Server 2008R2, and Windows 7 platforms. For a 32-bit color depth, the client device must be capable of displaying 32-bit color.

15-bit color depths are not supported. If this color depth is specified on the Terminal Server, SGD automatically adjusts the color depth to 8-bit.

Transport Layer Security

From Microsoft Windows Server 2003 and later, you can use Transport Layer Security (TLS) for server authentication, and to encrypt Terminal Server communications. SGD does not support the use of TLS.

Terminal Services Group Policies

For Windows Server 2003 and later, Terminal Services settings can be configured using Group Policy, as follows:

- Individual Windows Terminal Servers can be configured using a Local Group Policy Object (LGPO). In the Group Policy Object Editor, the Terminal Services settings are at: Local Computer Policy\Computer Configuration\Administrative Templates\Windows Components\Terminal Services
- Multiple Windows Terminal Servers can be configured using a Group Policy Object (GPO), linked to a domain or organizational unit (OU).

To improve performance, you might want to configure some or all of the following policies:

- **Keep-Alive Connections.** This policy specifies a keep alive time interval for the Terminal Services session. See also [“Keep Alive Configuration for Windows Terminal Servers” on page 144](#).
- **Limit Maximum Color Depth.** This policy controls the display color depth on client devices. See Microsoft Knowledge Base article 278502 for details of how to set this policy.

Keep Alive Configuration for Windows Terminal Servers

If you find that the connection between the SGD server and the Windows Terminal Server is being dropped unexpectedly, you might need to configure the keep alive mechanism for the Windows Terminal Server.

How to do this is described in Microsoft Knowledge Base article 216783.

Licensing Microsoft Windows Terminal Services

SGD does not include licenses for Microsoft Windows Terminal Services. If you access terminal server functionality provided by Microsoft operating system products, you need to purchase additional licenses to use such products. Consult the license agreements for the Microsoft operating system products you are using to determine which licenses you must acquire.

Terminal Services licensing is done using a CAL. A CAL is a license that allows a client to access the Windows Terminal Server. Depending on the licensing mode, a client can be either a *user*, or a *device*, or a combination of both.

Client license management for Microsoft Windows Terminal Services varies according to the client platform, as follows:

- **Windows platforms.** CALs for client devices that connect to the Terminal Server are allocated in accordance with Microsoft policy. CALs are stored in the Windows registry on the client device.
- **UNIX, Linux, or Mac OS X platforms.** CALs for client devices that connect to the Terminal Server are stored in a license pool in the datastore on the SGD server. Management of this license pool is done by the SGD Administrator, using the `tscal` command. See “Managing CALs From the Command-Line” on page 145.

Managing CALs From the Command-Line

You can use the `tarantella tscal` command to manage Microsoft Windows Terminal Services CALs for *non-Windows* client devices, as follows:

- To list the Terminal Services CALs currently reserved for non-Windows client devices, type the following:

```
$ tarantella tscal list
```

- To free a Terminal Services CAL for use by another non-Windows client devices, type the following:

```
$ tarantella tscal free
```

- To return all free Terminal Services CALs to the Windows License Server, type the following:

```
$ tarantella tscal return --free
```

Microsoft Windows Remote Desktop

Some editions of Microsoft Windows include a Remote Desktop feature that enables you to access a computer using Microsoft RDP. You can use SGD and Remote Desktop, for example, to give users access to their office PC when they are out of the office.

The supported platforms and features for Remote Desktop are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

Before introducing SGD, ensure that the Remote Desktop connection to the Microsoft Windows computer is working.

You configure SGD for use with Remote Desktop as follows:

- Create an application server object for each Microsoft Windows computer.
- Create a Windows application object for the Windows desktop application.

To ensure users access their own computer, you have to create separate Windows desktop application objects for each Microsoft Windows computer.

See [“Using My Desktop” on page 175](#) for details of how to run a full-screen desktop session, without displaying the SGD webtop.

Seamless Windows

With seamless windows, the Microsoft Windows application server manages the display of the application. This means an application’s windows behave in the same way as an application displayed on the application server, regardless of the user’s desktop environment. The window can be resized, stacked, maximized, and minimized. The Windows Start Menu and Taskbar are not displayed when using seamless windows.

Seamless windows are not suitable for displaying Windows desktop sessions. Use a kiosk or independent window instead.

The following are the conditions for using seamless windows:

- The SGD Enhancement Module for Windows must be installed on the application server.
- The Windows application object must be configured with a Window Type of seamless window.

If any of the above conditions are not met, SGD displays the Windows application in an independent window instead.

Notes and Tips on Using Seamless Windows

The following are some notes and tips on displaying applications in seamless windows:

- If an application is displayed in a seamless window, you can toggle between a seamless window and an independent window by pressing the Scroll Lock key. This does not work if the SGD Client is in Integrated Mode.
- Applications that have non-rectangular windows, for example, a media player with a customized skin, display in a rectangular window.

- On Windows client devices, seamless windows are not affected by the Cascade, Tile Windows Horizontally, or Tile Windows Vertically window commands.
- If a screen saver or the Windows Security dialog displays, the window automatically switches to an independent window. Unlocking the application automatically restores the window to a seamless window.
- If a seamless window application is resumed on a display that is larger or smaller in size than the original session, the application is displayed in an independent window.
- Each application displaying in a seamless window has its own RDP connection.
- Applications configured to display in seamless windows might not display correctly when accessed from a Gnome 2.0.0 Desktop. This is caused by an unpatched version of the Metacity Window Manager. The solution is to install the Gnome 2.0.0 Window Manager patch. Patch ID: 115780, available from the Sun Support Center at (<http://www.sun.com/support>).

Key Handling for Windows Terminal Services

You can configure how SGD handles keyboard presses on the client device in a Windows Terminal Services session, as follows:

- “Supported Keyboard Shortcuts for Windows Terminal Services” on page 147
- “The Windows Key and Window Management Keys” on page 148
- “Configuring Windows Keyboard Maps” on page 149

Supported Keyboard Shortcuts for Windows Terminal Services

SGD supports the following keyboard shortcuts for Windows Terminal Services sessions.

| Keyboard Shortcut | Description |
|-------------------|--|
| Ctrl-Alt-End | Displays the Windows Security dialog. |
| Alt-Page Up | Switches between windows, from left to right. |
| Alt-Page Down | Switches between windows, from right to left. |
| Alt-Insert | Cycles through windows, in the order they were opened. |
| Alt-End | Displays the Windows Start menu. |
| Alt-Delete | Displays the pop-up menu for the current window. |

| Keyboard Shortcut | Description |
|----------------------|---|
| Ctrl-Alt-Minus | Use the Minus (-) key on the numeric keypad. Places a snapshot of the active client window on the Windows Terminal server clipboard. Provides the same functionality as pressing Alt-PrintScrn on a local computer. |
| Ctrl-Alt-Plus | Use the Plus (+) key on the numeric keypad. Places a snapshot of the entire client window area on the Windows Terminal server clipboard. Provides the same functionality as pressing PrintScrn on a local computer. |
| Alt-Ctrl-Shift-Space | Minimizes the active window. Only applies for kiosk mode. |

The Windows Key and Window Management Keys

In SGD Windows Terminal Services sessions, the Windows key and keyboard shortcuts for managing windows can be sent either to the remote session or acted on locally. By default, they are acted on locally.

For Windows applications objects that are configured to display in kiosk mode, the Window Management Keys (`--remotewindowkeys`) attribute controls keyboard shortcut behavior. To send the Windows key and window management keys to the remote session, do either of the following:

- In the Administration Console, go to the Client Device tab for the Windows application object and select the Window Management Keys check box.
- Use the following command:

```
$ tarantella object edit --name obj --remotewindowkeys 1
```

If the Windows key and window management keys are sent to the remote session, use the key sequence Alt-Ctrl-Shift-Space to exit kiosk mode. This minimizes the kiosk session on the local desktop. Alternatively, to exit kiosk mode you can use the Kiosk Mode Escape (`--allowkioskescape`) attribute to enable a pull-down header for the application window. The pull-down header includes icons for minimizing and closing the kiosk session.

For Windows applications objects that are not configured to display in kiosk mode, you can force the Windows key to be sent to the remote session by using the `-windowskey` option for the SGD Remote Desktop Client. To send the Windows key to the remote session, do either of the following:

- In the Administration Console, go to the Launch tab for the Windows application object and type `-windowskey` on in the Arguments field.

- Use the following command:

```
$ tarantella object edit --name obj --protoargs "--windowskey on"
```

Configuring Windows Keyboard Maps

The process of configuring Windows keyboard maps in SGD is the same as that used for configuring keyboard maps for X applications. See also [“Keyboard Maps” on page 158](#).

Note – For Windows applications, the keyboard layout must be the same on the client device and the application server.

Returning Client Device Information for Windows Terminal Services Sessions

By default, when you run a Windows application through SGD using the Microsoft RDP protocol, the host name of the client device is returned in the `%CLIENTNAME%` environment variable for the Windows Terminal Services session. When you use a Sun Ray Desktop Unit (DTU) client device, the DTU ID is returned in the `%CLIENTNAME%` environment variable. The DTU ID is the hardware address of the Sun Ray.

The DTU ID can be used to specify the name of the client device in the `wcpwts.exp` login script. SGD uses this login script for all Windows applications that connect using the Microsoft RDP protocol.

The SGD Remote Desktop Client

The SGD Remote Desktop Client, also known as `ttatssc`, is a client program that handles the connection between the SGD server and the Windows Terminal Server.

The syntax for running `ttatssc` from the command line is as follows:

```
ttatssc [-options...] server.example.com
```

where *server.example.com* is the name of a Windows Terminal Server.

You can use the `ttatssc` to configure Windows Terminal Services sessions in the following ways:

- Configure attributes for the Windows application object. Some of the `ttatssc` command options are available as attributes for a Windows application object. These are indicated in the following table.
- Configure the Arguments (`--protoargs`) attribute of the Windows application object. Using this attribute, you can specify `ttatssc` command options used for a Windows application object.
- Edit the `wcpwts.exp` login script, and specify `ttatssc` command options. Any changes you make to this file are used for all Windows applications that connect using the Microsoft RDP protocol.

The following options are supported for the `ttatssc` command.

| Option | Description |
|--|--|
| <code>-application application</code> | The application to run in the Terminal Services session. |
| <code>-audioquality low medium high</code> | Sets the quality of the audio redirection. |
| <code>-bulkcompression on off</code> | Enable or disable data compression for the connection. |
| <code>-console</code> | Instead of starting a normal RDP session, connect to a console session. This option is available as the Console Mode (<code>--console</code>) attribute for a Windows application. |
| <code>-crypt on off</code> | Configures encryption for the connection. The default setting, <code>on</code> , gives the best user experience. |
| <code>-default depth</code> | Whether to let the Terminal Server set the default color depth of the X session. |
| <code>-desktop</code> | Whether to display a full screen desktop session. |
| <code>-dir working_dir</code> | Working directory for the Terminal Services session. This can be overridden by the application. This option is available as the Working Directory (<code>--workingdir</code>) attribute for a Windows application. |
| <code>-display X display</code> | The X display to connect to. |
| <code>-domain domain</code> | Domain on the Terminal Server to authenticate against. |
| <code>-keyboard language_tag</code> | Input locale. Specify an RFC1766 language tag. |
| <code>-name client name</code> | Name of the client device. |
| <code>-netbiosname name</code> | NetBIOS name for the client device. This is used for the redirected printer names on the Terminal Server. |
| <code>-noaudio</code> | Disables audio redirection. |
| <code>-nofork</code> | Do not run <code>ttatssc</code> as a background process. |

| Option | Description |
|--|--|
| <code>-noprintprefs</code> | Do not cache printer preferences. This option is available as the Printer Preference Caching (<code>--noprintprefs</code>) attribute for a Windows application. |
| <code>-opts file</code> | Read command options from a file. See “Using a Configuration File” on page 152 for details. |
| <code>-password password</code> | Password for the Terminal Services user. |
| <code>-perf disable wallpaper fullwindowdrag menuanimations theming cursorshadow cursorsettings</code> | <p>Disable display options, to improve performance. The available settings are:</p> <ul style="list-style-type: none"> • <code>wallpaper</code> – Disable the desktop wallpaper. This option is available as the Desktop Wallpaper (<code>--disablewallpaper</code>) attribute for a Windows application. • <code>fullwindowdrag</code> – Disable the option to show window contents when moving a window. This option is available as the Full Window Drag (<code>--disablefullwindowdrag</code>) attribute for a Windows application. • <code>menuanimations</code> – Disable transition effects for menus and tooltips. This option is available as the Menu Animations (<code>--disablemenuanimations</code>) attribute for a Windows application. • <code>theming</code> – Disable desktop themes. This option is available as the Theming (<code>--disabletheming</code>) attribute for a Windows application. • <code>cursorshadow</code> – Disable the mouse pointer shadow. This option is available as the Cursor Shadow (<code>--disablecursorshadow</code>) attribute for a Windows application. • <code>cursorsettings</code> – Disable mouse pointer schemes and customization. This option is available as the Cursor Settings (<code>--disablecursorsettings</code>) attribute for a Windows application. <p>To disable multiple display options, use multiple <code>-perf disable</code> options.</p> |
| <code>-perf enable fontsmoothing</code> | <p>Turns on font smoothing for text on the desktop.</p> <p>This option is available as the Font Smoothing (<code>--enablefontsmoothing</code>) attribute for a Windows application.</p> |
| <code>-port port</code> | RDP port to connect to on the Terminal Server. The default setting is 3389. |
| <code>-printcommand command</code> | <i>This option is deprecated.</i> |
| <code>-remoteaudio</code> | <p>Leaves audio at the Terminal Server.</p> <p>This option is available as the Remote Audio (<code>--remoteaudio</code>) attribute for a Windows application.</p> |
| <code>-sharedcolor</code> | Do not use a private color map. |
| <code>-size width height</code> | Display width and display height for the Terminal Services session, in pixels. |
| <code>-spoil</code> | <i>This option is deprecated.</i> |
| <code>-stdin</code> | Read command options from standard input. Used by the login scripts to pass command options to <code>ttatasc</code> . |

| Option | Description |
|--------------------------------------|--|
| <code>-storage data_dir</code> | <i>This option is deprecated.</i> |
| <code>-swmopts on off</code> | Enable local window hierarchy for applications that use seamless windows. Needed for some Borland applications. |
| <code>-timeout connect secs</code> | Timeout for connecting to the Terminal Server, in seconds. |
| <code>-timeout establish secs</code> | Timeout for establishing an RDP connection, in seconds. |
| <code>-uncompressed</code> | <i>This option is deprecated.</i> |
| <code>-user username</code> | User name for the Terminal Services user. |
| <code>-windowskey on off</code> | Whether to enable or disable Windows key for the Terminal Services session. The default setting is <code>on</code> . |

Using a Configuration File

A configuration file is a text file containing the `ttatssc` command-line options to be used for the connection. Each option must be on a separate line without the leading dash (-). The argument and its value are separated by whitespace. Use either single or double quotes to enclose any literal whitespace.

The escape character is `\`. The following escape sequences are supported:

- `\n` is a new line (0xA)
- `\r` is a carriage return (0xD)
- `\t` is a tab (0x9)
- `\\` is a literal `\`
- `\"` is a literal double quote not used for delimiting quoted arguments
- `\'` is a literal single quote not used for delimiting quoted arguments

The following is an example configuration file:

```
u "Indigo Jones"
p "Wh1teh411"
a "C:\\program files\\notepad.exe"
naples.indigo-insurance.com
```

Running Windows Applications on Client Devices

You can run a Windows application on a client device, instead of displaying it through SGD. If the application is not available on the client device, and the SGD Remote Desktop Client check box is selected, SGD tries to run it on the application server.

Applications that run on client devices are not resumable, *even if* the Application Resumability is configured.

The application must be installed in the same location on all client devices.

X Applications

This section describes how to configure X application objects.

This section includes the following topics:

- [“Configuring X Application Objects” on page 153](#)
- [“Supported X Extensions” on page 155](#)
- [“X Authorization” on page 155](#)
- [“X Fonts” on page 156](#)
- [“Keyboard Maps” on page 158](#)

Configuring X Application Objects

In the Administration Console, the configuration settings for X application objects are divided into the following tabs:

- **General tab** – These settings control the name and the icon used when creating links for users
- **Launch tab** – These settings control how the application is started and whether application sessions can be suspended and resumed
- **Presentation tab** – These settings control how the application is displayed to users
- **Performance tab** – These settings are used to optimize the performance of the application
- **Client Device tab** – These settings control how the user’s client device interacts with the application

The following table lists the most commonly used settings for configuring X application objects and how to use them.

| Attribute | Description |
|--------------------------|---|
| Name | The name that users see. |
| Icon | The icon that users see. |
| Application Command | <p>The full path to the application that runs when users click the link.</p> <p>The application must be installed in the same location on all application servers.</p> <p>The following are commonly used commands for desktop sessions:</p> <ul style="list-style-type: none"> • <code>/usr/dt/config/Xsession.jds</code> – For a Sun Java Desktop System desktop • <code>/usr/bin/gnome-session</code> – For a Gnome desktop • <code>/usr/bin/startkde</code> – For a K Desktop Environment (KDE) desktop <p>See also “Configuring Common Desktop Environment Applications” on page 198, and “Configuring VMS Applications” on page 201.</p> |
| Arguments for Command | <p>Any command-line arguments to use when starting the application.</p> <p>Note - Never specify a <code>-display</code> argument. This is set by SGD.</p> |
| Connection Method | The mechanism SGD uses to connect to the application server, for example telnet or ssh. |
| Number of Sessions | The number of instances of an application a user can run. The default is three. |
| Application Resumability | <p>For how long the application is resumable. The following options are available:</p> <ul style="list-style-type: none"> • Never – The application can never be resumed • During the User Session – The application keeps running and is resumable until the user logs out of SGD • General – The application keeps running for a time, controlled by the Timeout setting, after the user logs out of SGD, and can be resumed when the user next logs in |
| Session Termination | The circumstances when the SGD server ends the application session. |
| Window Type | <p>How the application is displayed to the user.</p> <p>Use Kiosk for full-screen desktop sessions. Selecting the Scale to Fit Window check box for the Window Size enables SGD to scale the application window to fit the client device display.</p> <p>Use Client Window Management to display the application as though it is running on the client device.</p> <p>For other window types, you must specify a Height and Width for the Window Size or select the Client’s Maximum Size check box.</p> |
| Color Depth | <p>The application’s color depth.</p> <p>SGD supports X applications with multiple color depths. So you can run an 8-bit application within a 24-bit desktop session by selecting 24/8-bit, for example</p> |

| Attribute | Description |
|---------------------------------|--|
| Application Load Balancing | How SGD chooses the best application server to run the application. See “Application Load Balancing” on page 362 for more details. |
| Hosting Application Servers tab | Use the Editable Assignments table to select the application servers, or group of application servers, that can run the application. The application must be installed in the same location on all application servers. |
| Assigned User Profiles tab | Use the Editable Assignments table to select the users that can see the application. Selecting Directory or Directory (light) objects enables you to give the application to many users at once. You can also use an LDAP directory to assign applications. See “LDAP Assignments” on page 127 . |

In addition to this configuration, you can also configure the following:

- Printing – See [“Printing” on page 223](#).
- Client drives – See [“Client Drive Mapping” on page 258](#).
- Audio – See [“Audio” on page 276](#).
- Copy and paste – See [“Copy and Paste” on page 287](#).

Creating X Application Objects on the Command Line

On the command line, you create an X application object with the `tarantella object new_xapp` command. You can also create multiple X application objects at the same time with the `tarantella object script` command. See [“Populating the SGD Organizational Hierarchy Using a Batch Script” on page 117](#).

X application objects can only be created in the `o=applications` organizational hierarchy.

Supported X Extensions

The supported X extensions are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

X Authorization

By default, SGD secures X displays using X authorization. This prevents users from accessing X displays that they are not authorized to access.

For information about troubleshooting X authorization for X applications, see [“Applications Fail To Start When X Authorization Is Enabled” on page 207](#).

X Fonts

SGD includes the standard X Window System fonts in compiled and compressed form (`.pcf.gz`), together with some additional fonts required by different UNIX systems. See [Fonts in X11R6.8.2](#) for details. The fonts are installed in the `/opt/tarantella/etc/fonts` directory.

The following X fonts and font directories are available with SGD.

| Directory | Description |
|-----------|--|
| 75dpi | Variable-pitch 75 dpi fonts. |
| 100dpi | Variable-pitch 100 dpi fonts. |
| andrew | Fonts from the Andrew toolkit, required by some IBM applications. |
| CID | A placeholder for CID-keyed fonts. If you want to add your own CID fonts for use with SGD, install them in this directory. |
| cyrillic | Cyrillic fonts. |
| encodings | A set of encoding files used by the Type1 and TrueType font handlers |
| hangul | Korean fonts. |
| hp | Fonts required by some Hewlett-Packard applications. |
| icl | Fonts required by some ICL applications. |
| misc | Fixed-pitch fonts, cursor fonts, and fonts for compatibility with older versions of X. |
| oriental | Kanji and other oriental fonts. |
| scoterm | Cursor fonts. |
| TTF | True Type fonts. |
| Type1 | PostScript™ Type 1 fonts. |

Using Your Own X Fonts

You can make your own X fonts available through SGD in the following ways:

- Use a font directory, see [“Using a Font Directory” on page 157](#)
- Use a font server, see [“Using a Font Server” on page 157](#)

After making the X fonts available, you must configure each SGD server in the array to use the fonts, see [“How to Configure SGD to Use Your Own X Fonts” on page 157](#).

Using a Font Directory

To use a font directory, copy your fonts in .pcf format to a directory on each SGD server in the array and include a `fonts.dir` file that maps filenames to X logical font descriptions.

The fonts can be gzipped (.pcf.gz), but fonts compressed using the `compress` command (.pcf.Z) are not supported. If a font is compressed in a .Z file, decompress it first before copying to the SGD server.

The following is an example line from a `fonts.dir` file:

```
COURBO10.pcf -Adobe-Courier-Bold-0-Normal-10-100-75-75-M-60-ISO8859-1
```

If your font directory does not include a `fonts.dir` file, you can use a program such as `mkfontdir`, which is available for most UNIX systems, to create one.

You can also include a `fonts.alias` file, that specifies aliases for the fonts in the directory. This file maps aliases to X logical font descriptions. For example:

```
variable *-helvetica-bold-r-normal-*-*-140-*
```

Using a Font Server

A font server is a program that makes fonts on a host available on the network. Font servers make font administration easier by centralizing fonts, reducing duplication.

To name a font server in a font path, you need to know the name of the font server and the port on which fonts are being served. For example, if the font server `boston` uses Transmission Control Protocol (TCP) port 7100, add the font path entry `tcp/boston:7100`.

▼ How to Configure SGD to Use Your Own X Fonts

Ensure that no users are logged in to the SGD server, and that there are no application sessions, including suspended application sessions, running on the SGD server.

1. In the Administration Console, go to the Secure Global Desktop Servers tab and select an SGD server.
2. Go to the Protocol Engines → X tab.

3. In the Font Path field, type the path to the directory containing your X fonts, or the location of the font server.

Each SGD server in the array can use a different font path. However, to avoid inconsistent display of applications, ensure that the same fonts, in the same order, are available to all SGD servers.

4. Click Save.

5. Restart the SGD server.

6. Check the validity of the font path.

Use the `xset` command to see if the font path is set.

```
$ xset q
```

Keyboard Maps

SGD uses a keyboard map, or *keymap*, file to process keyboard input for X applications. A keymap file contains a list of keys for the keyboard and the corresponding characters produced when you press the keys.

By default, an SGD server uses the keymap file corresponding to the locale specified by the Keyboard Map attribute on the Protocol Engines → X tab for the SGD server in the Administration Console.

The available locale settings are:

- **LANG Variable** – Use the locale of the SGD server. This is the value of the LANG environment variable on the SGD server.
- **Client's Input Locale** – Use the locale of the client device.
- **Select Custom Keyboard Map** – Specify your own keyboard map.

You can override the locale for a particular user, by setting the Keyboard Map (`--keymap`) attribute for the user profile object

To prevent an application from changing the default keyboard mappings, set the Keyboard Map: Locked (`--lockkeymap`) attribute for the application object.

Keymap files are located in the `/opt/tarantella/etc/data/keymaps` directory on the SGD server. This directory contains keymap files for the most common keyboard layouts. The keymap files in this directory have a file name beginning with *x*. For example, `xuniversal.txt` keymap file is used to map the keys of a Universal (English US) keyboard.

SGD uses the `/opt/tarantella/etc/data/keymaps/xlocales.txt` file to find the keymap file for the specified locale. This file maps locales to keymap files. For example, the `xlocales.txt` specifies the `xuniversal.txt` keymap file for a locale setting of `en_US`.

Character Applications

This section describes how to configure character application objects. Terminal emulator mappings are also discussed.

This section includes the following topics:

- [“Configuring Character Application Objects” on page 159](#)
- [“Terminal Emulator Keyboard Maps” on page 161](#)
- [“Terminal Emulator Attribute Maps” on page 166](#)
- [“Terminal Emulator Color Maps” on page 167](#)

Configuring Character Application Objects

You use a character application object if you want to give a VT420, Wyse 60, or SCO Console character application to users.

In the Administration Console, the configuration settings for character application objects are divided into the following tabs:

- **General tab** – These settings control the name and the icon used when creating links for users
- **Launch tab** – These settings control how the application is started and whether application sessions can be suspended and resumed
- **Presentation tab** – These settings control how the application is displayed to users
- **Performance tab** – These settings are used to optimize the performance of the application
- **Client Device tab** – These settings control how the user’s client device interacts with the application

The following table lists the most commonly used settings for configuring character application objects and how to use them.

| Attribute | Description |
|--------------------------|---|
| Name | The name that users see. |
| Icon | The icon that users see. |
| Application Command | The full path to the application that runs when users click the link. The application must be installed in the same location on all application servers. See also “Configuring VMS Applications” on page 201 for details of how to configure Virtual Memory System (VMS) character applications. |
| Arguments for Command | Any command-line arguments to use when starting the application. |
| Connection Method | The mechanism SGD uses to connect to the application server, for example telnet or ssh. |
| Number of Sessions | The number of instances of an application a user can run. The default is three. |
| Application Resumability | For how long the application is resumable. The following options are available: <ul style="list-style-type: none"> • Never – The application can never be resumed • During the User Session – The application keeps running and is resumable until the user logs out of SGD • General – The application keeps running for a time, controlled by a timeout value, after the user logs out of SGD, and can be resumed when the user next logs in |
| Window Close Action | What happens if the user closes the main application window using the Window Manager decoration. This attribute only applies for applications that use an Independent Window. |
| Window Type | How the application is displayed to the user. If Independent Window is selected, you must specify a Height and Width for the Window Size or select the Client’s Maximum Size check box. Specify the number of Columns and Lines to display in the terminal window. |
| Emulation Type | The type of character application to emulate. SGD supports VT420, Wyse 60, or SCO Console character applications. |
| Terminal Type | The application’s terminal type. Accept the default terminal type, or type you own type in the Custom field. |

| Attribute | Description |
|---------------------------------|--|
| Application Load Balancing | How SGD chooses the best application server to run the application. See “Application Load Balancing” on page 362 for more details. |
| Hosting Application Servers tab | Use the Editable Assignments table to select the application servers, or group of application servers, that can run the application. The application must be installed in the same location on all application servers. |
| Assigned User Profiles tab | Use the Editable Assignments table to select the users that can see the application. Selecting Directory or Directory (light) objects enables you to give the application to many users at once. You can also use an LDAP directory to assign applications. See “LDAP Assignments” on page 127 . |

To use and display the euro character, the terminal session must be capable of displaying 8-bit characters. To ensure this, enter the command `stty -istrip`. Also, the client device must be capable of entering the euro character.

Creating Character Application Objects on the Command Line

On the command line, you create a character application object with the `tarantella object new_charapp` command. You can also create multiple character application objects at the same time with the `tarantella object script` command. See [“Populating the SGD Organizational Hierarchy Using a Batch Script” on page 117](#).

Character application objects can only be created in the `o=applications` organizational hierarchy.

Terminal Emulator Keyboard Maps

The SGD terminal emulators associate keys on the user’s client keyboard with keys found on a real terminal. For each type of terminal emulator: SCO Console, Wyse 60, and VT420, there is a default keyboard mapping.

To change the default mappings or define additional mappings for a particular application, you can specify your own keyboard map file using an object’s Keyboard Map attribute.

Default Mappings

The emulators have built-in keyboard maps, which are equivalent to the following sample keymap files in the `/opt/tarantella/etc/data/keymaps` directory:

- `ansikey.txt` – For the SCO Console emulator
- `vt420key.txt` – For the VT420 emulator
- `w60key.txt` – For the Wyse 60 emulator

Note – Modifying these keyboard maps does not alter the default mappings used by SGD. The only way to do this is to specify a keyboard map in an application object's Keyboard Map attribute.

Creating a Keyboard Map

To create your own keyboard map, make a copy of one of the sample keyboard map files, and modify it to suit your application. You can modify a keyboard map in any text editor.

The format of a mapping is:

| |
|--|
| <i>ClientKeys</i> = <i>Translation</i> |
|--|

Where *ClientKeys* is the key, or keys, that the user presses on the client device, and *Translation* is the keystroke, or keystrokes, sent to the application on the application server. For example:

| |
|---------------|
| PageDown=Next |
|---------------|

With this mapping, when the user presses the Page Down key the emulator sends the keystroke Next to the application server.

If a particular key has a user-defined mapping, the default settings are overridden. If no user-defined mapping is present, the default mapping is sent to the application server.

You can send complete strings on a single key press, by surrounding the string in straight quotation marks. For example:

| |
|------------------|
| F1="hello world" |
|------------------|

To enter non-printable characters when mapping strings, use the code shown in the table below:

| Code | Meaning |
|------|---|
| \r | Carriage return |
| \n | Line feed |
| \" | Straight quotation marks |
| \e | Escape |
| \t | Tab |
| \nnn | The character with octal value <i>nnn</i> |
| \xHH | The character with hex value <i>HH</i> |

To specify modifier keys, such as Shift, Control, and Alt, in a mapping, separate the keys with the plus sign, +. For example:

```
Shift+NUMLOCK=INLINE
Shift+F1="\0330a"
Alt+Shift+Control+DELETE="\003[33~"
```

Key Names

The following are lists of key names that are valid in SGD keyboard maps. The [Client Device Keys](#) list shows the key names that represent keys on the user's client device. These are the keys that can be mapped to the emulator key names given in [Application Server Keystrokes](#), which are the keystrokes ultimately sent to the application on the application server.

Note – The default mappings between these key names are as found in the keyboard maps supplied with SGD. If a key is not in a keyboard map, then it is not mapped.

Client Device Keys

SGD supports the following keys on the user's client device:

- CURSOR_DOWN
- CURSOR_LEFT
- CURSOR_RIGHT

- CURSOR_UP
- DELETE
- END
- F1 to F12
- HOME
- INSERT
- KP0 to KP9
- KPADD
- KPDELETE
- KPDIVIDE
- KPENTER
- KPMULTIPLY
- KPSUBSTRACT
- NUMLOCK
- PAGEDOWN
- PAGEUP

Application Server Keystrokes

The following application server keystrokes are supported for *SCO Console* applications:

- CURSOR_DOWN
- CURSOR_LEFT
- CURSOR_RIGHT
- CURSOR_UP
- DELETE
- END
- F1 to F12
- HOME
- INSERT
- KP0 to KP9
- KPADD
- KPDIVIDE
- KPDOT
- KPMULTIPLY

- KPSUBSTRACT
- NUMLOCK
- PAGEDOWN
- PAGEUP

The following application server keystrokes are supported for *VT420* applications:

- CURSOR_DOWN
- CURSOR_LEFT
- CURSOR_RIGHT
- CURSOR_UP
- F1 to F20
- FIND
- INSERT
- KP0 to KP9
- KPCOMMA
- KPDOT
- KPENTER
- KPMINUS
- NEXT
- PF1 to PF4
- PREV
- REMOVE
- SELECT

The following application server keystrokes are supported for *Wyse 60* applications:

- CLRLINE
- CLRSCR
- CURSOR_DOWN
- CURSOR_LEFT
- CURSOR_RIGHT
- CURSOR_UP
- DELCHAR
- DELETE
- DELLINE
- F1 to F16
- HOME

- INSCHAR
- INSERT
- INSLINE
- KP0 to KP9
- KPCOMMA
- KPDELETE
- KPENTER
- KPMINUS
- NEXT
- PREV
- PRINT
- REPLACE
- SEND
- SHIFTHOME

Terminal Emulator Attribute Maps

Terminal emulator attribute maps enable you to change how character attributes such as bold or underline are displayed in the SGD terminal emulators. For example, you can specify that text that normally appears bold and underlined appears red in the SGD terminal emulators, but not red *and* bold and underlined.

SGD provides a default attribute map

`/opt/tarantella/etc/data/attrmap.txt`. This maps character attributes to the logical color `Color_15` (white). You can also create your own attribute map.

▼ How to Create Your Own Attribute Map

1. **As superuser (root), create a copy of**
`/opt/tarantella/etc/data/attrmap.txt` **to work on.**
2. **Edit the new file, so that character attributes map to your chosen colors.**
3. **Use the name of the file for the application object's Attribute Map attribute.**

Editing Character Attributes

The SGD attribute maps enable you to map the following attributes:

- Normal
- Bold
- Dim
- Blinking
- Underline
- Inverse

To map combinations of attributes, separate the attributes with the plus sign +, for example, `Bold+Underline`.

To display colors in the terminal emulators, SGD maps logical colors to RGB values. For example, the logical color `Color_9` maps to the RGB value `128 0 0` (red).

When mapping attributes to colors in your attribute map, specify the logical color name. For example:

- To change bold underlined text to red text:

```
Bold+Underline=Color_9
```

- To change inverse blinking text to light red text:

```
Inverse+Blinking=Color_1
```

For a complete list of logical color to RGB value mappings, refer to the comments in `attrmap.txt`.

You can change the default color mappings by editing the color map used by the terminal emulators. See [“Terminal Emulator Color Maps” on page 167](#).

Note – Wyse 60 terminals display only black and white colors. However, you can use the SGD Wyse 60 terminal emulator to display colors in your Wyse 60 applications. You can do this by using the attribute map to map character attributes in the Wyse 60 application to colors.

Terminal Emulator Color Maps

SCO Console (ANSI) and VT420 terminals support 16 colors. The SGD terminal emulator uses a color map to determine how these colors are presented in an application session.

Note – Wyse 60 terminals are monochrome. You can only switch the background and foreground colors, black and white, using the color map. However, you can map character attributes such as bold or underline to any of the 16 logical colors supported by the terminal emulator. See [“Terminal Emulator Attribute Maps” on page 166](#).

The color map maps the logical colors Color_0 through to Color_15, inclusive, to colors and the RGB values that SGD uses to represent those colors. The default mappings are as follows:

| Logical Color | Terminal Color | RGB Value Used by SGD |
|---------------|----------------|-----------------------|
| Color_0 | Black | 0 0 0 |
| Color_1 | Light red | 255 0 0 |
| Color_2 | Light green | 0 255 0 |
| Color_3 | Yellow | 255 255 0 |
| Color_4 | Light blue | 0 0 255 |
| Color_5 | Light magenta | 255 0 255 |
| Color_6 | Light cyan | 0 255 255 |
| Color_7 | High white | 255 255 255 |
| Color_8 | Gray | 128 128 128 |
| Color_9 | Red | 128 0 0 |
| Color_10 | Green | 0 128 0 |
| Color_11 | Brown | 128 128 0 |
| Color_12 | Blue | 0 0 128 |
| Color_13 | Magenta | 128 0 128 |
| Color_14 | Cyan | 0 128 128 |
| Color_15 | White | 192 192 192 |

To alter the defaults for a particular application, create your own color map, and specify it in the application object’s Color Map attribute.

A default text-format color map `/opt/tarantella/etc/data/colormap.txt` is provided.

Examples of Using Color Maps

- To make the color red brighter, change the RGB setting of Color_9 to 192 0 0.

- To change items that appear in light green to appear yellow, change the RGB setting of `Color_2` to `255 255 0`, the RGB value of yellow.
- One common color change is to switch the foreground and background colors between black and white. When you do this, you are not changing the foreground or background color as such, you are changing the way black (`Color_0`) and white (`Color_15`) are displayed. Therefore, if your application has a white background and you want to change it to a black background, change the value of `Color_15` to `0 0 0`, the RGB value of black.

Dynamic Launch

Dynamic launch is the term used to describe runtime changes that are applied when users start applications. Typically, the runtime changes enable users to select the application server that runs the application, or to choose the application that is started, or both. Configuring dynamic launch can involve dynamic application servers, dynamic applications, and client overrides.

This section covers the following topics:

- [“Dynamic Application Servers” on page 169](#)
- [“Dynamic Applications” on page 172](#)
- [“Client Overrides” on page 174](#)
- [“Using My Desktop” on page 175](#)
- [“Integrating SGD With Oracle VDI” on page 176](#)

Dynamic Application Servers

A *dynamic application server* is an object that represents a virtual server broker (VSB). SGD uses the VSB to obtain a list of application servers that can run an application. If a user can select an application server, a chooser page is displayed that enables the user to specify the application server.

Dynamic application servers are created on the Application Servers tab of the Administration Console, or by using the `tarantella object new_host --dynamic` command.

The VSB used for a dynamic application server is specified using the Virtual Server Broker Class (`--vsbclass`) attribute. Any configurable parameters for the VSB are specified using the Virtual Server Broker Parameters (`--vsbparams`) attribute.

You only need to create one dynamic application server object for each VSB you are using. In general, you only create multiple dynamic application servers for the same VSB if you want to pass different parameters to the VSB.

The following VSBs are supplied with SGD:

- **SGD.** This broker enables users to select an application server from the list of application servers assigned to the application.
See [“SGD Broker” on page 170](#) for more details about this VSB.
- **User-defined SGD.** This broker enables users to specify an application server, or to select from the list of application servers assigned to the application.
See [“User-Defined SGD Broker” on page 171](#) for more details about this VSB.
- **VDI.** This broker enables SGD to request a desktop from an Oracle Virtual Desktop Infrastructure (VDI) installation.
See [“VDI Broker” on page 172](#) for more details about this VSB.

You can develop your own VSB for use with SGD. Your broker must implement the `IVirtualServerBroker` interface that is included in the `sgd-webservices.jar` in the `/opt/tarantella/bin/java/com/sco/tta/services/proxy` directory.

Dynamic application servers are assigned to an application in the same way as conventional application servers, as described in [“How to Assign Application Servers to Applications” on page 126](#).



Caution – Only assign one dynamic application server to an application.

Dynamic application servers override the normal SGD mechanisms for application load balancing because they enable users to choose where an application is run. To prevent application servers from becoming overloaded, you can use the attributes on an application server object to filter the application servers shown on the chooser page. See [“Dynamic Application Servers and Load Balancing” on page 364](#) for more details.

When dynamic application servers are used, entries in the SGD password cache are usually stored using the dynamic application server as well as the application server. But this can depend on the VSB and configured client overrides.

SGD Broker

The SGD broker lists the application servers that are assigned to an application object. The dynamic application server itself is not listed.

The user experience when using the SGD broker is as follows:

- When the user starts an application, a chooser page is displayed that lists the application servers that can run the application. The user must select an application server, and click the Start button to run the application.
- If only one application server can run the application, the chooser page is not displayed. The application is automatically run on the application server.
- If the application is a dynamic application, the user selects both the application and the application server.

On the command line, the fully-qualified class name (`--vsbclass`) for the SGD broker, is `com.tarantella.tta.webservices.vsbim.SGDBroker`.

The SGD broker does not have any configurable parameters.

When you install SGD, a default dynamic application server object called `o=appservers/cn=SGD Broker` is created automatically. This dynamic application server is used with the My Desktop application. See [“Using My Desktop” on page 175](#) for details.

User-Defined SGD Broker

The User-defined SGD broker lists the application servers that are assigned to an application object, and also enables users to specify the name or IP address of any application server. Users can run applications on application servers that do not have a corresponding application server object in the local repository. The dynamic application server itself is not listed.

The user experience when using the User-defined SGD broker is as follows:

- When the user starts an application, a chooser page is displayed. The user either types the name or IP address of an application server in the field provided on the chooser page, or selects an application server from the list, and clicks the Start button to start the application.
- If no SGD-configured application servers are available to the user, the user must type the name or IP address of an application server.
- If the application is a dynamic application, the user must select both the application and the application server.

On the command line, the fully-qualified class name (`--vsbclass`) for the User-defined SGD broker is `com.tarantella.tta.webservices.vsbim.UserDefinedSGDBroker`.

The User-defined SGD broker has a one optional parameter. If you specify `createAppserver` for the Virtual Server Broker Parameters (`--vsbparams`) attribute, SGD automatically creates new application server objects for any user-specified application servers that do not already exist in the local repository.

VDI Broker

The VDI broker enables SGD to request a desktop from a Oracle VDI installation. See [“Integrating SGD With Oracle VDI” on page 176](#) for details.

The user experience when using the VDI broker is as follows:

- When the user starts an application, a login page is displayed. The user must type a VDI user name and password and click the Login button.

By default the user’s SGD credentials are tried, so the login page might not be displayed. SGD can also be configured to cache the VDI credentials, so that the authentication page only displays once for each user.

- If the user is assigned multiple VDI desktops, whether a dedicated desktop or desktop pool, a chooser page is displayed that lists the available desktops. The user must select a desktop, and click the Start button to run the application.
- If the user is assigned a single VDI desktop or desktop pool, the chooser page is not displayed. The VDI desktop is run automatically.

On the command line, the fully-qualified class name (`--vsbclass`) for the VDI broker is `com.sun.sgd.vsbim.SunVDIVirtualServerBroker`.

The VDI broker has a one mandatory parameter, the host name of a VDI server. When you create an dynamic application server for the VDI broker, specify `localhost` for the Virtual Server Broker Parameters (`--vsbparams`) attribute.

Dynamic Applications

A *dynamic application* represents one or more application objects. When the user starts a dynamic application, a chooser page is displayed that enables the user to select an application to run.

A dynamic application object consists of a set of mappings between type strings and SGD application objects. For example, you could create a dynamic application to enable users to choose between a Windows desktop session or Linux desktop session. Such a dynamic application might use the mappings shown in the following table.

| Type | Application |
|---------|-----------------------------------|
| windows | o=applications/cn=Windows Desktop |
| linux | o=applications/cn=Linux Desktop |

Dynamic applications are created on the Applications tab of the Administration Console, or by using the `tarantella object new_dynamicapp` command. See [“How to Create a Dynamic Application” on page 173](#).

Type-application mappings for dynamic applications can be configured on the Mappings tab for the dynamic application object, or with the `tarantella object add_mapping` and `tarantella object remove_mapping` commands.

The type is a string that is specified when a mapping is added. Because the type is displayed on the chooser page, generally the type identifies the type of application. But it can be any unique string you want.

Dynamic applications are assigned to users in the same way as conventional applications, as described in [“Publishing Applications” on page 124](#). You can assign multiple dynamic applications to a user.

When you install SGD, a default dynamic application object called `o=applications/cn=My Desktop` is created automatically and is used for the My Desktop application. See [“Using My Desktop” on page 175](#) for details.

▼ How to Create a Dynamic Application

Ensure that the applications that you want to map to the dynamic application already exist.

1. In the Administration Console, go to the Applications tab.

2. Create the dynamic application object.

a. Select an object in the organizational hierarchy.

Use the navigation tree to select a directory object to contain the dynamic application.

b. In the content area, click New.

The Create a New Object window is displayed.

c. In the Name field, type the name of the dynamic application.

The name you type is used for the link on the webtop.

d. Ensure that the Dynamic Application option is selected and click Create.

The Create a New Object window closes and the content area is updated with the new object.

3. Configure the dynamic application.

a. Click the View New Object link.

The General tab for the dynamic application object is displayed.

b. (Optional) Change the Icon for the dynamic application.

The icon is used on the webtop.

c. Click the Mappings tab.

d. In the Editable Mappings table, click Add.

The Add a New Mapping window is displayed.

e. In the Mapping Type field, type a string to identify the mapping.

The string can be anything. The string is displayed on the chooser page that is displayed to users. Usually the type identifies the type of application.

f. Select the check box next to an application object.

Use the Navigation tree to browse for a directory object that contains the application.

You can only select application objects.

g. Click Add.

The Add a New Mapping window closes and the Mappings tab is updated with the new mapping.

h. Repeat steps d to g to create further mappings.

4. Assign the dynamic application to users.

Dynamic applications are assigned to users in the same way as conventional applications. See [“Publishing Applications” on page 124](#).

Client Overrides

Client overrides are a comma-separated list of options used to configure dynamic launch. By default, the client override to configure support for dynamic applications and dynamic application servers is enabled.

You use the following command to configure client overrides:

```
$ tarantella config edit \  
--tarantella-config-applaunch-allowclientoverrides opt ...
```

where *opt* is a comma-separated list. The following table lists the available options.

| Option | Description |
|---------------------|--|
| appserver_pw | Read password cache entries for application servers. |
| array_pw | Read the password cache for users' SGD passwords. |
| dynamic | Enable support for dynamic applications and dynamic application servers. |
| dynamicappserver_pw | Read password cache entries for dynamic application servers. |
| false | Disable all client overrides. |
| true | Enable all client overrides. |

For example, to disable all client overrides, use the following command:

```
$ tarantella config edit \  
--tarantella-config-applaunch-allowclientoverrides false
```

For example, if you are integrating with Oracle VDI and you want SGD to read the password cache for the users' VDI credentials, use the following command:

```
$ tarantella config edit \  
--tarantella-config-applaunch-allowclientoverrides dynamic,dynamicappserver_pw
```

Using My Desktop

My Desktop enables users to log in and display a full-screen desktop without displaying a webtop.

To be able to use My Desktop, a user must be assigned an application object called My Desktop (cn=My Desktop).

A default My Desktop object (o=applications/cn=My Desktop) is created automatically when SGD is installed. This object is a dynamic application object that has the type-application mappings shown in the following table.

| Type | Application |
|---------|-----------------------------------|
| windows | o=applications/cn=Windows Desktop |
| unix | o=applications/cn=Unix Desktop |

By default, this object runs the default desktop application available on the SGD server, as configured for the `o=applications/cn=Unix Desktop` application object. The windows type-application mapping for My Desktop is configured to run a Windows desktop application. However to be able to use Windows Desktop, you must first assign at least one application server object to the `o=applications/cn=Windows Desktop` application object. If you do this, users are prompted to choose which desktop application to run. See [“Dynamic Applications” on page 172](#) for more details.

You can reconfigure the default My Desktop object to run any application you want, but it works best with full-screen desktop applications. If users require different desktop applications, you can create additional My Desktop objects as required.

Users access My Desktop from their webtop or by using the My Desktop URL, at `http://server.example.com/sgd/mydesktop`, where *server.example.com* is the name of an SGD server. This Uniform Resource Locator (URL) displays the SGD Login page. Once the user has logged in, selected an application and application server (if configured), the desktop session is displayed. After the user has logged in, the browser window can be closed.

Alternatively, users can click the My Desktop link on the SGD web server Welcome Page, at `http://server.example.com`.

Users can be assigned any number of applications, but the My Desktop URL only gives users access to the My Desktop application. To use the My Desktop URL, a user must be assigned only one application called My Desktop.

If the user has paused print jobs, they see a message in the browser window when they log in which enables them to resume printing. You can disable this feature by setting the following value in the `mydesktop/index.jsp` file, which is located in the `/opt/tarantella/webserver/tomcat/tomcat-version/webapps/sgd/` directory.

```
boolean promptForPrintResume=false
```

Integrating SGD With Oracle VDI

SGD includes a VDI broker that enables you to give users access to desktops provided by an Oracle VDI server.

The supported versions of Oracle VDI are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

To integrate SGD with VDI, SGD and VDI must be installed on the same host.

Integrating SGD with Oracle VDI involves the following configuration steps:

1. Create a Windows application object for use with VDI.

SGD connects to VDI using RDP and so you must use a Windows application object.

You could create an application object specifically for VDI integration, or you could adapt the My Desktop application. See [“Using My Desktop” on page 175](#).

2. Create a dynamic application server for the VDI broker.

See [“How to Create a Dynamic Application Server for the VDI Broker” on page 177](#).

For information about dynamic application servers and the VDI broker, see [“Dynamic Application Servers” on page 169](#).

3. Assign the VDI dynamic application server to the VDI Windows application

Dynamic application servers are assigned to applications in the same way as conventional application servers, as described in [“How to Assign Application Servers to Applications” on page 126](#).



Caution – Ensure that *only* the VDI dynamic application server is assigned to the application. Remove any conventional application server assignments.

4. Assign the VDI Windows application object to users.

See [“Publishing Applications” on page 124](#).

5. (Optional) Configure the client override to enable the caching of passwords.

By default, SGD prompts users for credentials every time they connect to a VDI desktop using the VDI broker. See [“Client Overrides” on page 174](#).

▼ How to Create a Dynamic Application Server for the VDI Broker

1. In the Administration Console, go to the Application Servers tab.

2. Create a dynamic application server object for the VDI broker.

- a. Select an object in the organizational hierarchy.

Use the navigation tree to select a directory object to contain the dynamic application server.

- b. In the content area, click New.

The Create a New Object window displays.

SSH Support

SGD works with SSH version 2 or later. Because of SSH version compatibility problems, use the same major version of SSH, either version 2 or version 3, on all SGD hosts and application servers.

SGD can automatically detect that SSH is installed on the SGD host if SSH is installed in one of the following directories:

- /usr/local/bin
- /usr/bin
- /usr/sbin
- /usr/lbin
- /bin
- /sbin

If you want to run the SSH client from a different location, or you want to specify particular command-line arguments for the client, see [“Configuring the SSH Client” on page 179](#) for details.

To connect to an application server using SSH, the following must be true:

- SSH must be installed on the SGD host and on the application server
- The application object’s Connection Method attribute must be ssh

Configuring the SSH Client

When using SSH with SGD, you can configure the command-line arguments used by the SSH client. The arguments can be configured globally, for individual applications, or a combination of both.

You configure the *global options* for the SSH client by setting the `TTASSSHCLIENT` environment variable, see [“How to Set Global SSH Client Options” on page 180](#) for details. Use the global SSH client configuration in the following situations:

- SSH is not installed in one of the default locations
- To use the same SSH client command-line arguments for all applications

You configure the *application options* for the SSH client by configuring the SSH Arguments attribute for the application object, see [“How to Set Application SSH Client Options” on page 180](#) for details.

You can combine the global and application SSH client configuration to set the path to the SSH client and set the command-line arguments.

Note – If you do this, any global command-line arguments are ignored.

The following table shows the effect of global and application configuration on the `ssh` command used.

| Global Configuration | Application Configuration | SSH Command Used |
|--------------------------|---------------------------|--|
| [none] | [none] | <code>ssh -l user@host</code> |
| [none] | -X | <code>ssh -X -l user@host</code> |
| <code>/usr/ssh -X</code> | [none] | <code>/usr/ssh -X -l user@host</code> |
| <code>/usr/ssh -X</code> | <code>-p port</code> | <code>/usr/ssh -p port -l user@host</code> |

▼ How to Set Global SSH Client Options

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

1. **Log on as superuser (root) on the SGD host.**

2. **Stop the SGD server.**

3. **Set the `TTASSHCLIENT` environment variable.**

Include the full path to the SSH client program and any required command-line arguments. For example:

```
# TTASSHCLIENT="/usr/local/bin/ssh -q -X"; export TTASSHCLIENT
```

Note – If you only want to set command-line arguments for the SSH client, you have to include the full path to the SSH client program, *even if* the SSH program is in a location where SGD can detect it.

4. **Restart the SGD server.**

▼ How to Set Application SSH Client Options

1. **In the Administration Console, go the Applications tab and select the application.**

2. **Go to the Launch tab.**

3. **Ensure that the `ssh` option is selected for the Connection Method.**

4. In the SSH Arguments field, type the SSH arguments you want to use for the application.
5. Click Save.

Enabling X11 Forwarding for X Applications

To display X applications using SGD using an SSH connection, you must enable X11 forwarding. See [“How to Enable X11 Forwarding” on page 181](#).

As a fallback, you can enable the Allow SSH Downgrade (`--allowsshdowngrade`) attribute on X application objects. If this attribute is enabled and X11 forwarding is not working or not configured, SGD tries to display the application using a regular unsecured X11 connection. Depending on your configuration, users might be prompted to accept the downgrade. The following table shows the effect of enabling the Allow SSH Downgrade attribute.

| X11 Forwarding Configured | X11 Forwarding Working | Allow SSH Downgrade Attribute Enabled | What Happens When Users Start the Application |
|---------------------------|------------------------|---------------------------------------|--|
| Yes | Yes | Yes | Application starts, SSH connection used |
| Yes | Yes | No | Application starts, SSH connection used |
| Yes | No | Yes | User prompted to allow downgrade to X11 connection |
| Yes | No | No | Application does not start |
| No | Not applicable | No | Application does not start |
| No | Not applicable | Yes | Application starts, X11 connection used |

If an X11 connection is used, SGD sets the `DISPLAY` variable and X authorization cookie in the normal way. The SSH connection is only used for application authentication and for starting the application.

▼ How to Enable X11 Forwarding

1. Log in as superuser (root) on the SGD host.

2. Configure the SSH daemon.

Edit the `sshd_config` file and add the following line:

```
X11Forwarding yes
```

3. Configure the SSH client.

Do either of the following:

- Edit the `ssh_config` file and include the following lines:
`ForwardAgent yes`
`ForwardX11 yes`
- Configure the SSH client to use the `-x` command-line argument.
See [“Configuring the SSH Client” on page 179](#) for details.

4. Restart the SSH daemon.

Using SSH and the X Security Extension

SGD supports the X Security extension. The X Security extension only works with versions of SSH that support the `-Y` option. For OpenSSH, this is version 3.8 or later. You enable the X Security extension by configuring the application objects individual applications as follows:

▼ How to Enable the X Security Extension

1. In the Administration Console, go to the Applications tab and select the application.
2. Go to the Launch tab.
3. Ensure that the `ssh` option is selected for the Connection Method.
4. Select the X Security Extension check box.
5. Click Save.

Using SSH and X Authorization

If SSH connections fail when X authorization is enabled, you might have to run the SSH daemon in IPv4-only mode because SGD might not support the X Security extension used on your server. You enable IP version 4 mode by editing your system SSH configuration file.

For example, on Red Hat Enterprise Linux, edit the `/etc/sysconfig/sshd` file and add the following line:

`OPTIONS="-4"`

You must restart the SSH daemon after making this change.

Using Advanced SSH Functions

Certain SSH functionality, such as client keys, requires that the SSH client process runs as a privileged user. However, for security reasons, the SGD server processes and the SSH client process run as a non-privileged user.

To use advanced SSH functions, you must make the SGD `ttassshelper` application a `setuid` root process. You do this by running the following commands as superuser (root) on each SGD server in the array:

```
# chown root /opt/tarantella/bin/bin/ttassshelper
# chmod 4510 /opt/tarantella/bin/bin/ttassshelper
```



Caution – If you make these changes, you must protect your SGD servers from unauthorized access.

Known Limitation With Client Keys

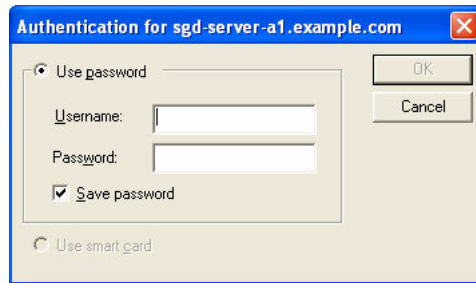
If you are using the SSH client keys functionality, users might be prompted for a user name and password when they start an application. Users are prompted because SGD needs to know the user name to use for the SSH connection. Although users are also prompted for a password, the password is not actually used. Users are only prompted for a user name and password if they do not have an entry in the password cache for the application server, or if the password cache is disabled. If users are prompted, they only need to provide a user name. The password field can be left blank.

Application Authentication

When a user clicks a link to start an application, the login script configured for the application connects to the application server, handles the authentication process, and starts the application.

The Execution Protocol Engine is the SGD component that runs the login script. The login script authenticates the user to the application server by submitting a user name and password stored in the SGD application server password cache. If there is a problem with the user's credentials, SGD displays an Application Authentication dialog as follows:

FIGURE 4-1 Screen Capture of the SGD Application Authentication Dialog



The Application Authentication dialog enables users to enter their credentials and store them in the application server password cache so that they are not prompted when they next run an application on that application server.

Users can also force SGD to display the Application Authentication dialog by holding down the Shift key when they click an application's link on the webtop.

Note – You cannot use the Shift key in this way when the SGD Client is in Integrated mode.

This section includes the following topics:

- [“Login Scripts” on page 184](#)
- [“Configuring Application Authentication” on page 185](#)
- [“The Application Server Password Cache” on page 185](#)
- [“Input Methods and UNIX Platform Applications” on page 188](#)
- [“Adding Support for System Prompts in Different Languages” on page 189](#)
- [“Using RSA SecurID for Application Authentication” on page 189](#)

Login Scripts

SGD uses login scripts to handle the connection to the application server, to run the application, and to perform additional tasks.

Typically a login script performs the following tasks:

- Logs in to the application server, prompting the user for a password if necessary.
- Sets environment variables. These are the variables specified by the Environment Variables attribute on the Launch tab for the application object.
- Starts any window manager programs. These are the programs specified by the Window Manager attribute on the Presentation tab for the application object.

- Starts an input method if one is required (UNIX platform applications only).
- Runs the command to start the application.

The login script takes into account the differences between application servers, and checks for any errors that might occur during the login process. If an error is encountered that cannot be handled, control is passed back to the user.

The SGD login scripts are designed to be as universal and robust as possible. However, you might need to cope with an unusual scenario. For example, if you have a system prompt that is not catered for, you can add it to the list of prompts recognized by the script.

The login scripts supplied with SGD also contain commands and procedures that you can use to customize the display of the Application Authentication dialog, for example by adding your own labels for the Username and Password fields.

If you need to customize a login script, make a copy of an SGD login script and work on the copy. Do not modify the standard SGD login scripts. [Appendix E](#) contains detailed reference information about SGD login scripts.

Configuring Application Authentication

In the Administration Console, the attributes on the Global Settings → Application Authentication tab control application authentication. These attributes allow you to configure the following:

- Whether to automatically try the user's SGD user name and password when logging in to an application server if these details have been cached
- What action to take if the user's application server password has expired
- Whether to log in to a Microsoft Windows application server using a smart card
- When to display the Application Authentication dialog, what the default settings are on the dialog, and whether users can change them

The Application Server Password Cache

By default, SGD stores the user names and passwords used to run applications in its application server password cache. SGD also stores the user names and passwords used to log in to SGD.

Note – SGD cannot store the user name and password used to log in to SGD if the user is authenticated with third-party authentication.

For *Windows applications*, the Terminal Server handles the authentication process. No information is returned to SGD indicating whether authentication succeeds or fails, and the details remain in the SGD password cache whether correct or incorrect.

Managing the Application Server Password Cache

In the Administration Console, you can manage the application server password cache as follows:

- **The Caches → Passwords tab** – This tab enables you to manage any entry in the password cache
- **The Passwords tab for user profile objects** – This tab enables you to manage password cache entries for the selected user profile
- **The Passwords tab for application server objects** – This tab enables you to manage password cache entries for the selected application server
- **The Passwords tab for dynamic application server objects** – This tab enables you to manage password cache entries for the selected dynamic application server

On the command line, you manage the application server password cache with the `tarantella passcache` family of commands.

You can use the Administration Console and the command line to list and delete entries in the password cache. You can also create entries in the password cache. With the `tarantella passcache` command, you can populate the password cache with a batch script.

Each entry in the password cache involves the following elements:

- **A user name** – The user name for the application server
- **A password** – The password for the application server
- **A resource** – The application server, dynamic application server, or domain name for which the password is cached
- **A user identity** – The identity of the user that “owns” the entry in the password cache

Note – The user’s SGD password can also be stored in the password cache.

Security and the Password Cache

Entries in the application server password cache are encrypted with an encryption key. When starting applications, the passwords are decrypted as they are needed.

By default, the encryption key used for the password cache never changes. You can configure SGD to generate a new encryption key for the password cache whenever an SGD server restarts. In the Administration Console, go to the Global Settings → Security tab and select the New Password Encryption Key check box. Alternatively, use the following command:

```
$ tarantella config edit --security-newkeyonrestart 1
```

Existing entries in the password cache are re-encrypted with the new key.

Windows Domains and the Password Cache

When SGD caches a user's password for a Microsoft Windows application server, the password cache entry is created using the Windows domain name.

The domain name can be specified using the Domain Name attribute on application server objects, Windows application objects, or user profile objects. Users can also specify a domain name on the Application Authentication dialog.

When a user starts an application, SGD goes through the following process to establish the domain name and password cache entry to use:

1. Check if a domain name is set on the application server object.

If a domain name is set, SGD searches the password cache for an entry for the user identity.

If there is no domain name, or there is no entry in the password cache, move to step 2.

2. Check if a domain name is set on the application object.

If a domain name is set, SGD searches the password cache for an entry for the user identity.

If there is no domain name, or there is no entry in the password cache, move to step 3.

3. Check if the user typed a domain name type when they logged in to SGD.

If you are using Windows Domain authentication, users can specify a domain name when they log in to SGD. They do this by typing a user name in the format domain\name, for example example\rusty.

If a domain name is set, SGD searches the password cache for an entry for the user identity.

If there is no domain name, or there is no entry in the password cache, SGD displays the Application Authentication dialog.

The Application Authentication dialog has an NT Domain field that enables users to set the domain name. This field is automatically completed if the Domain Name attribute is set for the application server or application object, or if the domain is cached in the password cache. If the Domain Name attribute is set only on the user profile object, the NT Domain field is not completed.

To force users to specify a domain when they start a Windows application for the first time, you must ensure that the Domain Name attribute is blank for the user profile object, the application server object, and the application object.

If a user's SGD password is also their Windows domain password, the domain name and password can be cached if the following are true:

- SGD must be configured to save the user's SGD user name and password in the password cache. SGD does this by default.
- The Domain Name must be set on the user profile object.

Note – If the user is authenticated using Windows Domain authentication, you do not need to set the Domain Name attribute on the user profile object as the domain name can be inferred.

Input Methods and UNIX Platform Applications

An input method is a program or operating system component that enables users to enter characters and symbols not found on their keyboard.

By default, SGD runs an IM for all locales, apart from C and POSIX.

To change the IM configuration, you can edit variables the `vars.exp` login script. The variables are as follows:

- `IM_runByDefault` – This variable controls whether an IM is enabled (variable set to 1), or is disabled (variable set to 0) by default. The default is 1.
- `IM_localeNotList` – If `IM_runByDefault` is set to 1, SGD runs an IM for all locales, except the locales listed in the `IM_localeNotList` variable.
- `IM_localeList` – If `IM_runByDefault` is set to 0, SGD only runs an IM for the locales listed in the `IM_localeList` variable.

SGD uses the following environment variables to determine the locale `TTA_PREFERREDLOCALE`, `TTA_HOSTLOCALE`, and `LANG`. See [“Login Script Variables” on page 866](#).

Adding Support for System Prompts in Different Languages

By default, the login scripts supplied with SGD support English system prompts on application servers. SGD Administrators can add support for system prompts in other languages.

To do this, you edit the `vars.exp` login script and add a translation for each of the English prompts defined. The `vars.exp` login script is located in the `/opt/tarantella/var/serverresources/expect` directory on the SGD server. You do not need to translate every prompt, only the prompts that are different to the English ones. The file contains examples to help you get started. You can also provide translations for the variables, strings, and error message section to match the client or user locale.

In the Administration Console, configure the General tab → Prompt Locale attribute for your application server objects, to match a locale defined in `vars.exp`.

Using RSA SecurID for Application Authentication

SGD supports RSA SecurID authentication for X and character applications.

To use SecurID authentication, ensure that users can log in to the application server using SecurID before introducing SGD. When you are ready to use SecurID authentication, configure the application object to use the `securid.exp` login script.

When logging in to an application server that uses SecurID authentication, users enter a user name and password. When they click OK, they are prompted for a passcode.

In the Administration Console, go to the Global Settings → Application Authentication tab and deselect the Password Cache Usage check box. This prevents SGD from using SGD login details when logging in to the application server.

Tips on Configuring Applications

This section contains tips on configuring applications and documents for use with SGD. This section includes the following topics:

- [“Starting an Application or Desktop Session Without Displaying a Webtop” on page 190](#)

- [“Using Multihead Or Dual Head Monitors” on page 191](#)
- [“Improving the Performance of Windows Applications” on page 194](#)
- [“Improving the Performance of Java Desktop System Desktop Sessions or Applications” on page 195](#)
- [“Documents and Web Applications” on page 196](#)
- [“Creating a Virtual Classroom” on page 196](#)
- [“Configuring Common Desktop Environment Applications” on page 198](#)
- [“Configuring VMS Applications” on page 201](#)
- [“3270 and 5250 Applications” on page 202](#)

Starting an Application or Desktop Session Without Displaying a Webtop

With SGD, you can start a single application or a full-screen desktop session without displaying the webtop. You can do this in any of the following ways:

- **Use My Desktop** – My Desktop enables users to log in and display an application without displaying a webtop. Typically this is used for full-screen desktop applications. See [“Using My Desktop” on page 175](#) for details.
- **Use Integrated mode** – Integrated mode adds links for running applications to the desktop Start or Launch menu on the client device. See [“Integrated Mode” on page 320](#) for more details.
- **Use SGD web services** – Develop your own application for starting a single application. See [“Using SGD Web Services” on page 190](#) for more details.

Using SGD Web Services

You can use SGD web services to develop your own *application launcher*, to start a single application from a URL. You can use this method to start an application from a bookmark or a favorite. SGD provides an example application that shows what is possible using SGD web services.

The URL for using the SGD example application is:

```
http://server.example.com/sgd/launcher.jsp?o=application&u=username&p=password&e=true|false
```

where *server.example.com* is the name of an SGD server.

The URL has the following parameters:

| Parameter | Description |
|----------------------------|--|
| <code>o=application</code> | The name of the application object. This does not have to be a fully-qualified name. |
| <code>u=username</code> | The user name to use to log in to SGD. |
| <code>p=password</code> | The password to use to log in to SGD. |
| <code>e=true false</code> | <code>true</code> means display an edit page where users can override some of the application attributes. <code>false</code> means do not display edit page. |

Note – All of the parameters are optional.

For example, the following URL starts the Write-o-Win application using the configuration for the application object defined in the Administration Console.

```
http://server.example.com/sgd/launcher.jsp?o=Write-o-Win&u=indigo&p=purple&e=false
```

Using Multihead Or Dual Head Monitors

You can use multihead or dual head monitors with SGD. However, if any of your X applications are configured with a Window Type (`--displayusing`) setting of Client Window Management or Kiosk, you might have to change the application and monitor configuration to be able to use multiple monitors.

See also [“Configuring X Application Objects” on page 153](#).

To configure multiple monitors to work with X applications that have a Window Type setting of Client Window Management or Kiosk, perform the following configuration steps:

1. Disable shared resources.
 - See [“Disabling Shared Resources” on page 192](#).
2. Configure the correct desktop size.
 - See [“Configuring the Correct Desktop Size” on page 192](#).
3. Set up the monitors.
 - See [“Setting Up the Monitors” on page 193](#).

Disabling Shared Resources

SGD enables similar applications to share resources, to reduce memory overhead. This feature must be disabled for any applications that you want to display using multiple monitors.

In the Administration Console, go to the Performance tab for the application that is displayed on multiple monitors and deselect the Share Resources Between Similar Sessions check box.

Alternatively, use the following command:

```
$ tarantella object edit --name obj --share false
```

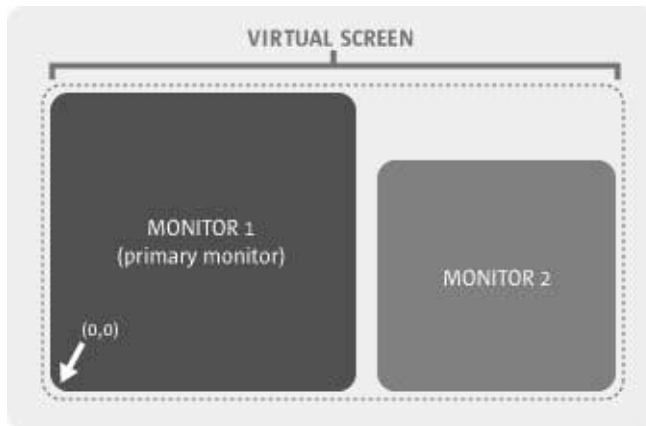
Repeat the configuration for each application that is displayed on multiple monitors.

Configuring the Correct Desktop Size

You must ensure that the SGD server sends the client device enough space to display all the monitors on the desktop.

Note – This increases the amount of memory used on the client device and on the SGD server.

You must configure SGD so that it sends the size of the entire desktop area to the client device, and not just the size of the primary monitor. The size of the entire desktop area is shown by the “virtual screen” in the following diagram.



For example, if the dimensions of Monitor 1 in the diagram are 1200 x 768 and the dimension of Monitor 2 are 800 x 600, then the desktop size that needs to be configured is 2000 x 768.

Configuring Desktop Size for Client Window Management Applications

In the Administration Console, go to the Protocol Engines → X tab for the SGD server. For Client Window Size, type the dimensions in pixels of the virtual screen in the Maximum Height and Maximum Width fields.

Alternatively, use the following command:

```
$ tarantella config edit --array \  
--xpe-cwm-maxwidth pixels --xpe-cwm-maxheight pixels
```

Repeat this configuration for each SGD server in the array.

Configuring Desktop Size for Kiosk Mode Applications

X applications can be displayed in kiosk mode on a dual head monitor.

You configure kiosk mode display features with the <KioskArea> entry in the <localsettings> section of the client profile, profile.xml on the client device. If the <localsettings> section is not present in the client profile, create a new section.

The <KioskArea> entry defines the screen area used by kiosk mode. The available values are as follows:

- virtual – Uses the virtual screen size. Both monitors are used.
- 0 – Uses the primary monitor only. This is the default value.
- 1 – Uses the secondary monitor only.

For example, to span the kiosk mode display across both monitors:

```
<KioskArea>virtual</KioskArea>
```

Setting Up the Monitors

Set up the monitors so that all the secondary monitors are to the right of the primary monitor, as shown in the previous diagram.

You have to do this because the X server cannot handle negative screen coordinates.

Improving the Performance of Windows Applications

When using Windows Terminal Services, the performance of Windows applications might be poor. This is caused by using animation effects and other presentation settings in the Windows session. Performance is affected because these features require more screen updates and can greatly increase the bandwidth used. The problem is more severe on slower connections.

The causes of these problems can include the following:

- Animated mouse cursors
- Mouse cursor shadows
- Screensavers
- Animated icons in the notification area
- Animated images in programs
- Animated wallpaper
- Images used as wallpaper
- Font smoothing

By default, the SGD Remote Desktop Client, `ttatssc`, enables all of these features except for font smoothing.

You can turn off presentation features by configuring one or more attributes for the Windows application object. The attributes you can configure are listed in the following table.

| Attribute | Description |
|-------------------|---|
| Cursor Settings | Disables mouse pointer schemes and customizations. |
| Cursor Shadow | Disables the shadow on the mouse pointer. |
| Desktop Wallpaper | Disables the desktop wallpaper. Disabling the wallpaper can reduce the amount of data that is updated when users move items around the desktop. |
| Font Smoothing | Enables font smoothing for text in the application. |
| Full Window Drag | Disables the option to show the contents of a window while it is moved. |
| Menu Animations | Disables transition effects for menus and tooltips. |
| Theming | Disables desktop themes. |

See also [“The SGD Remote Desktop Client” on page 149](#).

Improving the Performance of Java Desktop System Desktop Sessions or Applications

This section includes some tips on how to get the best user experience when using SGD with Java Desktop System.

You can improve performance of Java Desktop System desktop sessions and applications in the following ways:

- Configure the X application object for Java Desktop System
- Disable some default Java Desktop System settings

Configuring the X Application Object for Java Desktop System

For a Java Desktop System desktop session or application, ensure that you specify the correct command path for the X application object. Set the Application Command (`--app`) attribute to `/usr/dt/config/Xsession.jds`. Using a path of `/usr/bin/gnome-session` results in some Java Desktop System configuration parameters being lost and gives a poorer user experience.

To improve the display of animated effects, you can configure performance settings for the X application object. See [“An Application’s Animation Appears ‘Jumpy’” on page 215](#).

Disabling Default Java Desktop System Settings

The performance of Java Desktop System desktop sessions and applications can be affected by animation effects and other default desktop settings. Performance is affected because these features require more screen updates and can greatly increase the bandwidth used. The problem is more severe on slower connections.

Performance can often be improved by turning off, or modifying, some of the following Java Desktop System desktop features:

- Anti-aliased fonts
- Large fonts
- Login screen, splash screen, About screen, and Logout screen
- Animations
- Desktop applets
- Showing window contents when dragging
- Desktop wallpaper

■ Themes

Documents and Web Applications

A document object can refer to any URL. This can be any document on the web, including StarOffice™ software documents, or Adobe™ Acrobat files. A document can also refer to a web application.

As it is the user's *client device* that actually fetches the URL, firewalls, or other security measures can prevent a user from accessing a document.

You can use SGD to access web applications. A web application is simply a web page, or any URL, that requires the user to supply a user name and password for access. To give users access to a web application, you create a document object that links to the URL of the web application.

Unlike passwords for application servers, SGD cannot cache the user names and passwords for accessing web applications. However, you can configure web server authentication, so that users can access SGD from a web application without having to log in again. Alternatively, you can authenticate SGD users to the web application.

When accessing web applications, use a secure (HTTPS) web server, so that all communication is encrypted using SSL before transmission.

Creating a Virtual Classroom

This section describes how to configure application objects for use in a *virtual classroom*.

You can use SGD shadowing to create a virtual classroom, where the *pupils* in the classroom shadow an application being demonstrated by a *teacher*.

To be able to do this, you have to create a teacher's application object and a classroom application object.

The teacher must start their application first, then the pupils start their classroom application to shadow the teacher. The classroom can only shadow Windows applications or X applications.

Only one person can use the teacher's application at any one time. If more than one person starts the teacher's application, the classroom shadows the application that was started last. For this reason, only give the teacher's application to one user. If you have several teachers, create separate application objects for each of them.

The classroom application must have a color depth of at least 16-bit. The display size of the classroom application must be at least the size of the teacher's application. For the best results, use an independent window for the classroom.

When the teacher starts their application, information is stored on the SGD server about which application can be shadowed by the classroom. This information is *not* copied to the other members of the array. This means that if the classroom application is started on a different SGD server to the teacher's application, the classroom application fails because the information about which application can be shadowed is not available. You can use load balancing groups to guarantee that the teacher and classroom applications are started on the same SGD server. You must set the load balancing group for the application server *and* the SGD server. Otherwise, only use classroom shadowing in an SGD array containing a single SGD server.

See also [“Using Shadowing to Troubleshoot a User's Problem”](#) on page 214.

▼ How to Create the Teacher's Application Object

1. In the Administration Console, create a new Windows application object or X application object.
2. Go to the Launch tab and type one of the following in the Login Script field:
 - `unixclass.exp` – If the application is an X application
 - `winclass.exp` – If the application is a Windows application
3. Click Save.
4. Configure any other settings you want for the teacher's application.
5. Click the Hosting Application Servers tab and select the application servers that can run the application.
6. On the Assigned User Profiles tab, assign the teacher's user profile to the application.

▼ How to Create the Classroom Application Object

1. In the Administration Console, create a new X application object.

Note – The classroom application is an X application, even if the teacher's application is a Windows application.

The General tab is displayed.

2. Go to the Launch tab and configure the application as follows:

- a. **In the Application Command field, type**
`/opt/tarantella/bin/bin/ttashadow.`
 - b. **In the Arguments For Command field, type** `-readonly -silent -pointer $SHADOWDISPLAY.`
 - c. **For the Connection Method, select the telnet option.**
Select this option even if telnet is disabled for the SGD server.
 - d. **In the Login Script field, type** `pupil.exp.`
 - e. **In the Environment Variables field, configure the MYCLASS variable.**
The format is `MYCLASS="name_of_teacher's_application"`. For example,
`MYCLASS=".../_ens/o=applications/ou=Finance/cn=XClaim"`.
3. **Click Save.**
 4. **Go to the Presentation tab.**
 5. **For Color Depth, select 16-bit - thousands of colors and click Save.**
 6. **Configure any other settings you want for the classroom application.**
 7. **Go to the Hosting Application Servers tab and select the application servers that can run the application.**
The `ttashadow` application is only available on servers where SGD is installed.
 8. **Go to the Assigned User Profiles tab, assign the user profiles of all users in the class to the classroom application.**

Configuring Common Desktop Environment Applications

The configuration required for Common Desktop Environment (CDE) applications depends on whether you want to run a desktop session or an individual application.

For CDE desktop sessions configured with a Connection Method of `ssh`, problems can occur when a user tries to exit from the CDE session. The CDE session might hang, making it impossible to log out normally from the system. See [“Using CDE and SSH” on page 200](#).

Configuring a CDE Desktop Session

To run a CDE desktop session through SGD, create an X application object with the settings shown in the following table.

| Attribute | Setting |
|-----------------------------|---|
| Application Command | The full path to the <code>Xsession</code> application, for example, <code>/usr/dt/bin/Xsession</code> . On the command line, use <code>--app pathname</code> . |
| Keep Launch Connection Open | Select the Enabled check box. On the command line, use <code>--keepopen true</code> . |
| Session Termination | Select Login Script Exit from the list. On the command line, use <code>--endswhen loginscript</code> . |
| Window Type | Select Kiosk from the list. On the command line, use <code>--displayusing kiosk</code> |
| Window Size | Select the Scale to Fit Window check box. Use this setting only if users suspend and resume the application on displays of different sizes. On the command line, use <code>--scalable true</code> |

Configuring a CDE Application

To run a CDE application directly, rather than from the CDE Front Panel, create an X application object with the settings shown in the following table.

| Attribute | Setting |
|-----------------------------|--|
| Application Command | The full path to the application you want to run. On the command line, use <code>--app pathname</code> . |
| Keep Launch Connection Open | Deselect the Enabled check box. On the command line, use <code>--keepopen false</code> . Note - This is the default value for this attribute. |

| Attribute | Setting |
|---------------------|---|
| Session Termination | Select No Visible Windows from the list. On the command line, use <code>--endswhen nowindows</code> . Note - This is the default value for this attribute. |
| Window Type | Select Client Window Management from the list. On the command line, use <code>--displayusing clientwm</code> |
| Window Manager | Type the following in the field: <code>/usr/dt/bin/dtwm -xrm "Dtwm*useFrontPanel: false" -xrm "Dtwm*ws0*backdrop*image: none"</code> On the command line, use <code>--winmgr '/usr/dt/bin/dtwm -xrm "Dtwm*useFrontPanel: false" -xrm "Dtwm*ws0*backdrop*image: none" '</code> |

Using CDE and SSH

For CDE desktop sessions configured with a Connection Method of `ssh`, problems can occur when a CDE desktop user tries to exit from the CDE session. The CDE session might hang, making a proper logout from the system impossible.

The CDE session displays a `TT_ERR_NO_MATCH` error message.

The workaround for this issue is as follows:

- Log in to the CDE host as superuser (root) and type the following commands:

```
# mkdir /etc/dt
# mkdir /etc/dt/config
# cp /usr/dt/config/sessionetc /etc/dt/config
# cp /usr/dt/config/sessionexit /etc/dt/config
# cp /usr/dt/config/sys.dtpofile /etc/dt/config
# chgrp bin /etc/dt/config
# chmod 555 /etc/dt/config/*
# chown bin:bin /etc/dt/config/*
```

- Add the following lines to the `/etc/dt/config/sessionetc` file:

```
if [ "$SSH_TTY" != "" ]
then
SSHPTY='echo $SSH_TTY | cut -c6-15\'
ps -ef | grep -v grep | grep $SSHPTY | grep Xsession | awk '{print $3}' >
/var/dt/tmp/$DTUSERSESSION/sshd_pid
fi
```


- Add the following lines to the `/etc/dt/config/sessionexit` file:

```
if [ -f /var/dt/tmp/$DTUSERSESSION/ssh_d_pid ]
then
/bin/kill -HUP `/bin/cat /var/dt/tmp/$DTUSERSESSION/ssh_d_pid`
/bin/rm /var/dt/tmp/$DTUSERSESSION/ssh_d_pid
fi
```

- Add the following line to the `/etc/dt/config/sys.dtprofile` file:

```
dtstart_session[0]="/usr/local/bin/ssh-agent /usr/dt/bin/dtsession"
```

Configuring VMS Applications

You can use SGD to access X applications and character applications on a VMS application server.

To configure SGD to access applications on a VMS server, you have to perform the following configuration steps:

1. Configure the login script used for the application.
 - See [“Configuring the Login Script Used for the Application” on page 201](#).
2. Configure the transport variable in the login script.
 - See [“Configuring the Transport Variable in the Login Script” on page 202](#).
3. Disable X security.
 - See [“Disabling X Security” on page 202](#).

Configuring the Login Script Used for the Application

The login script used for the X application or character application must be configured.

In the Administration Console, go to the Applications → Launch tab for the application object you want to configure.

In the Login Script box, type one of the following:

- `vms.exp` – If `telnet` or `ssh` is selected as the Connection Method
- `vmsrexec.exp` – If `rexec` is selected as the Connection Method

Alternatively, use the following command:

```
$ tarantella object edit --name obj --login vms.exp | vmsrexec.exp
```

Configuring the Transport Variable in the Login Script

By default, the `vms.exp` or `vmsrexec.exp` login scripts set the transport variable to `TCPIP`. This setting is correct for Digital Transmission Control Protocol/Internet Protocol (TCP/IP) stacks, including Ultrix Communications Extensions (UCX).

If you need to change this variable, edit the transport variable setting in the login script. The transport variable is set by the following entry in the login script:

```
set transport "TCPIP"
```

The login scripts are located in the `/opt/tarantella/var/serverresources/expect` directory.

Disabling X Security

To use VMS X applications, you must disable X security in SGD. This is because VMS does not support X authorization.

In the Administration Console, go to the Global Settings → Security tab and deselect the X Authorization for X Display check box.

Alternatively, use the following command:

```
$ tarantella config edit --security-xsecurity 0
```

3270 and 5250 Applications

SGD uses the third-party emulator application, TeemTalk for Unix, for 3270 and 5250 applications. See the TeemTalk for Unix User's Guides supplied with SGD for details.

The first time a user runs the 3270 or 5250 emulator, the `tta3270.nv` configuration file is created in the user's home directory on the SGD server.

Troubleshooting Applications

This section describes some typical problems that users might have with their applications, and how to fix them.

To troubleshoot problems with starting and ending applications, see the following:

- [“An Application Does Not Start” on page 203](#)
- [“An Application Exits Immediately After Starting” on page 207](#)
- [“Applications Fail To Start When X Authorization Is Enabled” on page 207](#)
- [“Applications Disappear After About Two Minutes” on page 209](#)
- [“An Application Session Does Not End When the User Exits an Application” on page 210](#)
- [“Users Can Start Applications With Different User Names and Passwords” on page 212](#)
- [“Using Windows Terminal Services, Users Are Prompted for User Names and Passwords Too Often” on page 213](#)

To troubleshoot problems with using applications, see the following:

- [“Using Shadowing to Troubleshoot a User’s Problem” on page 214](#)
- [“A Kiosk Application Is Not Appearing Full-Screen” on page 215](#)
- [“An Application’s Animation Appears ‘Jumpy’” on page 215](#)
- [“Font Problems with X Applications” on page 215](#)
- [“Display Problems With High Color X Applications” on page 216](#)
- [“Clipped Windows With Client Window Management Applications” on page 218](#)
- [“Emulating a Sun Keyboard” on page 218](#)
- [“Display Update Issues When Shadowing Over a Low Bandwidth Connection” on page 220](#)
- [“Troubleshooting Mouse Drag Delay Issues” on page 220](#)
- [“Incorrect Time Zone Name Shown in Windows Applications” on page 221](#)

An Application Does Not Start

If an application does not start when a user clicks a link, first check the configuration of the application object, see [“Checking the Configuration of the Application Object” on page 204](#).

If this does not resolve the problem, check the launch details or the log files for a launch error message, see [“Checking the Launch Details and Error Logs” on page 204](#).

If users cannot log in to SGD or start applications, do a warm restart of the SGD servers by running the following command:

```
# tarantella restart sgd --warm
```

Checking the Configuration of the Application Object

Check the configuration of the application object in the Administration Console.

First, check the Hosting Application Servers tab for the application object. You must specify at least one application server to run the application. Check that the listed application servers are available.

Next, check the Launch tab for the application object. Check the attributes shown in the following table.

| Attribute | What to Check |
|-----------------------|--|
| Application Command | <p>Does the command contain the full path name of the application's executable?</p> <p>For Windows application objects, does the command also include the correct filename extension?</p> <p>Does the path name point to a Windows shortcut? If so, change it to the full path name of the application itself.</p> <p>Is the application installed in the same location on every application server listed in the Hosting Application Servers tab?</p> |
| Arguments for Command | Are the command arguments correct? |
| Connection Method | For X and Character application objects, is the selected Connection Method appropriate for the type of application server? |
| Login Script | <p>Is a login script set?</p> <p>Is the login script appropriate for the application type?</p> |
| Environment Variables | Are all the environment variables required for the application set correctly? |

If the application object is configured correctly, check that the application itself actually runs on all the application servers.

Checking the Launch Details and Error Logs

When an application fails to start, SGD displays an error message in the details area of the Connection Progress dialog. The error message is output to the SGD Client log file (`tcc.txt`) in the user's home directory.

The error messages are also output to the following log files:

- `/opt/tarantella/var/log/execpePID_error.log`

This file contains log output from the Execution Protocol Engine process.

- `/opt/tarantella/var/log/launchhelperPID_error.log`

This file contains additional log output if the connection method for the application object is SSH.

The error messages have the following form:

```
ErrorMessage
Script process-id exited with code error-code and signal signal
```

The *ErrorMessage* and *error-code* can be used to troubleshoot problems. The following are the most common error messages:

- ErrApplicationServerTimeout

See [“Troubleshooting ErrApplicationServerTimeout Errors” on page 206](#)

- ErrApplicationServerLoginFailed

See [“Troubleshooting ErrApplicationServerLoginFailed Errors” on page 206](#).

For a complete list of error messages and codes and troubleshooting information, see [“Login Script Error Messages” on page 878](#).

If the launch details or log files show error messages such as “Failed to find xauth” or “Attempt to run xauth failed”, see [“Applications Fail To Start When X Authorization Is Enabled” on page 207](#).

Increasing the Log Output

If you are still unable to resolve the problem, you can increase the amount of information that is output to the log files. To do this, you amend the log filters for the Execution Protocol Engine, and, *for X and character applications only*, enable debugging in the login script.

To amend the log filter for the Execution Protocol Engine, use the following command:

```
$ tarantella config edit \
--tarantella-config-execpeconfig-logfilter \
"execpe/*/*" "pem/*/*" "launchhelper/*/*"
```

To enable debugging in the login scripts, edit the following files:

- The login script configured for the application object.

Remove the comment mark (#) from the start of the `startdebug` line.

The login script is usually either `unix.exp`, `securid.exp`, `vms.exp`, `unixclass.exp`, or `pupil.exp`.

- `procs.exp`

Insert a comment mark (#) at the start of the `stopdebug` line.

When you have resolved the problem, remember to reset the log filter for the Execution Protocol Engine back to the default, and disable logging in the login scripts. Use the following command to reset the log filter:

```
$ tarantella config edit \  
--tarantella-config-execpeconfig-logfilter \  
"execpe/*/*error" "pem/*/*error" "launchhelper/*/*error"
```

Troubleshooting ErrApplicationServerTimeout Errors

If starting an application results in an `ErrApplicationServerTimeout` error, this usually means a login script has timed out before it can log the user in.

You can fix this by increasing the login script timeouts. See [“Login Script Timeouts” on page 874](#) for details of the available timeouts.

When changing the timeouts, increase the Expect timeouts first. If the application still fails to start, one of the client timers might be too short. If the application is particularly slow to start, increase all the client timers.

Increasing the login script timeouts slows down application start times. Only change the timeouts if you are experiencing problems, and tune the timeouts to the capabilities of the application server.

Note – None of the timeouts, apart from the Execution Protocol Engine timeout, apply when running a Microsoft Windows application.

Troubleshooting ErrApplicationServerLoginFailed Errors

If starting an application results in an `ErrApplicationServerLoginFailed` error, the login script failed to log into the application server.

Check that you can log into the application server manually.

If you can log in, check that the application server’s system prompt is recognized by the login script. Unusual system prompts are a common reason for this failure, and can be caused by the following:

- System prompts in a language other than English
- Message-of-the-day (`/etc/motd`) or issue messages (`/etc/issue`)
- The user’s login profile is configured to run a menu

By default, SGD supports English system prompts on application servers. Administrators can add support for system prompts in other languages. See [“Adding Support for System Prompts in Different Languages” on page 189](#) for details.

If you are using the standard SGD login scripts, check the system prompts defined in the `vars.exp` login script.

If a message-of-the-day or a menu causes a login script to fail, you must configure the login script to handle this situation. Alternatively, contact Technical Support for help.

The login script might also be timing out. If you see `“echo SYNC”` in the launch details or log files, and the system prompt ends in the normal way with `$`, `%`, `#`, or `>`, try increasing the `timeouts(prelogin)` value in the `vars.exp` login script. See [“Expect Timeouts” on page 874](#) for details.

An Application Exits Immediately After Starting

Users might see this problem with X applications. The solution is to keep open the network connection used to start the application.

In the Administration Console, select the Keep Launch Connection Open check box on the Launch tab for the application object.

Alternatively, use the following command:

```
$ tarantella object edit --name obj --keepopen true
```

Applications Fail To Start When X Authorization Is Enabled

In a default SGD installation, X authorization is enabled. If there are any problems with X authorization, users cannot start applications. If applications fail to start because of X authorization, the message `“Failed to find xauth”` or `“Attempt to run xauth failed”` displays in the application launch details and in the log files.

Use the following checklist to establish why X authorization causes application to fail to start. If this does not resolve the issue, check the log files as described in [“Checking the Launch Details and Error Logs” on page 204](#).

Is X authorization installed on the application server?

For SGD to be able to use X authorization, `xauth` must be installed on every application server.

If `xauth` is not installed, you must either install it or disable the use of X authorization for every application. To disable X authorization, deselect the X Authorization for X Display check box on the Global Settings → Security tab in the Administration Console.

Can SGD find the `xauth` binary?

If the message “Failed to find `xauth`” displays in the application launch dialog or log files, SGD cannot find the `xauth` binary. By default, SGD searches the following locations for the `xauth` binary:

- `/usr/bin/X11/xauth`
- `/usr/X/bin/xauth`
- `/usr/X11R6/bin/xauth`
- `/usr/bin/X/xauth`
- `/usr/openwin/bin/xauth`
- `/usr/bin/xauth`

If the `xauth` binary is in a different location, you must add its location to the `/opt/tarantella/var/serverresources/expect/vars.exp` login script. Look for the line beginning “`set xauthcmds`”.

Note – If the `xauth` binary is only in one location, you can speed up application launches by removing the other locations from the `vars.exp` login script.

Does the user have a UNIX system account on the application server?

When the user starts an application, the X Protocol Engine process generates a cookie and stores it in the `.Xauthority` file in the user’s home directory on the application server. The cookie is used to validate whether or not the user has permission to connect to the X display.

If the user does not have a home directory, the cookie cannot be stored in the user’s `.Xauthority` file and so the user cannot be validated.

You can do any of the following:

- Create a home directory for the user on the application server
- Disable X authorization

Deselect the X Authorization for X Display check box on the Global Settings → Security tab in the Administration Console. Alternatively, use the following command:

```
$ tarantella config edit --security-xsecurity 0
```

- Edit configuration files on the application server, so that the cookie is stored in a temporary directory.

Add the following lines to the `/etc/profile` file on the application server:

```
XAUTHORITY=/tmp/.Xauthority.$LOGNAME
export XAUTHORITY
```

Create the following SSH daemon configuration file, `/etc/ssh/sshrd`, on the application server:

```
HOME=/tmp
XAUTHORITY=$HOME/.Xauthority.$USER
export XAUTHORITY

if read proto cookie && [ -n "$DISPLAY" ]
then
    if [ `echo $DISPLAY | cut -c1-10` = 'localhost:' ]
    then
        # X11UseLocalhost=yes
        echo add unix:`echo $DISPLAY |
        cut -c11-` $proto $cookie
    else
        # X11UseLocalhost=no
        echo add $DISPLAY $proto $cookie
    fi | /usr/openwin/bin/xauth -q -
fi
```

Applications Disappear After About Two Minutes

If users find that their applications disappear unexpectedly after about two minutes, a proxy server might be timing out the connections. Proxy servers drop a connection after a short period of time if there is no activity on the connection.

SGD sends keepalive packets to keep the connection open and by default this is every 100 seconds. If applications are disappearing, you can increase the frequency at which keepalive packets are sent to keep the connection open.

In Administration Console, go to the Global Settings → Communication tab for the application object and set the AIP Keepalive Frequency to a smaller value than the default, for example, 60.

Alternatively, use the following command:

```
$ tarantella config edit --sessions-aipkeepalive secs
```

Note – You must restart every SGD server in the array for changes to this attribute to take effect.

An Application Session Does Not End When the User Exits an Application

If an application does not end when a user closes down the application, first check the configuration of the application object, see [“Checking the Session Termination Setting” on page 210](#).

To troubleshoot problems with Windows applications, see [“Windows Applications Do Not Close Down” on page 210](#).

To troubleshoot problems with UNIX desktop sessions, see [“UNIX Desktop Sessions Do Not Close Down After Logging Out” on page 212](#).

Checking the Session Termination Setting

In the Administration Console, go to the Launch tab for the application object and check the value of the Session Termination attribute.

If No Visible Windows is selected, the application session ends when there are no visible windows.

Windows Applications Do Not Close Down

When running an application on a Microsoft Windows Terminal Server, closing the application does not always result in the session closing. This is because the SGD Enhancement Module for Windows is still running. The solution is to configure the SGD Enhancement Module to ignore certain system processes so that it closes. To do this, edit the System processes value for the HKEY_LOCAL_MACHINE\Software\Sun Microsystems, Inc.\Enhancement Module for Windows key in the registry on the application server. This value is a string that contains a

comma-separated list of .exe binaries that the SGD Enhancement Module can ignore. You must amend this value by listing the processes that were running when the session failed to close. To do this, open Task Manager, while you have a session that has failed to close, and go to the Processes tab. Make a list of all the .exe processes that are running. Do not include the following processes:

- clipsrv.exe
- conime.exe
- csrss.exe
- EventLog.exe
- lmsvcs.exe
- lsass.exe
- MsgSvc.exe
- nddeagnt.exe
- netdde.exe
- NETSTRS.EXE
- os2srv.exe
- proquota.exe
- rdpclip.exe
- screg.exe
- smss.exe
- spoolss.exe
- ttaswm.exe
- userinit.exe
- wfshell.exe
- win.com
- winlogon.exe

If you are running a single application session, you might find that the session still does not exit, even after editing the System processes registry setting. To force the session to exit, amend the Logoff application sessions setting for the HKEY_LOCAL_MACHINE\Software\Sun Microsystems, Inc.\Enhancement Module for Windows key and change the DWORD value to 1.

UNIX Desktop Sessions Do Not Close Down After Logging Out

X applications configured as a full-screen desktop session might not close down when you log out from the application using the desktop Start menu. For example, desktop applets can remain open after logging out from the application. The application then has to be closed down using the pull-down header in the application window.

The workaround is to modify the Application Command (`--app`) attribute for the X application object, so that the `TTA_SESSION_STATE` property is removed from the root window when the application is closed down.

The following example shows how to modify the Application Command for a Java Desktop System desktop session.

- Create a simple shell script as follows:

```
#!/bin/sh
/usr/dt/config/Xsession.jds
/usr/openwin/bin/xprop -root -remove TTA_SESSION_STATE
```

- Save the script to a location on the SGD server, for example `/usr/local/bin/launch.sh`.
- In the Administration Console, go to the Launch tab for the X application object and edit the Application Command field to use the path to the script.

Alternatively, use the following command:

```
$ tarantella object edit --name objname --app "/usr/local/bin/launch.sh"
```

Users Can Start Applications With Different User Names and Passwords

By default, users can force SGD to display the Application Authentication dialog by holding down the Shift key when they click an application's link on the webtop. This enables users to start applications with different user name and passwords.

Note – You cannot use Shift-click with the SGD Client when it is in Integrated mode.

You can disable the Shift-click behavior. In the Administration Console, go to the Global Settings → Application Authentication tab and deselect the On Shift-Click check box. Alternatively, use the following command:

```
$ tarantella config edit --launch-showauthdialog system
```

Disabling the Shift-click behavior means that the Application Authentication dialog only displays when there is a problem with the password or there is no password.

Using Windows Terminal Services, Users Are Prompted for User Names and Passwords Too Often

If you are using Windows Terminal Services, users might be prompted for a user name and password by SGD or by the Terminal Server.

SGD Prompts the User

If SGD always prompts the user for a user name and password, the problem is usually caused by a missing domain name. If the user has no entries in the password cache that have a domain name, the Application Authentication dialog is displayed.

To fix this problem, the domain name must be provided when saving details in the password cache. You must do this even if the application server is not part of a domain.

The easiest way to configure the domain name is with the Domain Name attribute on the application server object or the application object. Users can also provide their own domain names in the Application Authentication dialog. See [“Windows Domains and the Password Cache” on page 187](#).

Terminal Server Prompts the User

SGD sends user name and password information to Windows Terminal Services to authenticate the user. If the authentication fails, Windows prompts the user again. No information is returned to SGD indicating whether authentication succeeds or fails, and the details remain in the SGD password cache whether correct or incorrect.

The user might have saved the wrong user name, password or domain name in the password cache.

To fix, the user must press Shift when clicking the link to start, the application. This displays the Application Authentication dialog, and the user can correct their user name, password, and domain name. Alternatively, delete the user's entry in the password cache so that SGD prompts the user the next time they start the application.

The Terminal Server might also be configured to always prompt for a password when a user logs in. Microsoft Windows 2000 Server does this by default, but Microsoft Windows Server 2003 and later does not. See [“Authentication Settings” on page 141](#) for details on how to change this behavior.

Using Shadowing to Troubleshoot a User's Problem

If a user is having difficulty with an application, you can use the Administration Console to find the user's application session and then *shadow* it. Shadowing enables the user and an SGD Administrator to see and use the application simultaneously.

To find a user's application session, go to the Application Sessions tab for their user profile object. Alternatively, go to the Application Sessions tab for the application object. This lists the users who are currently running the application.

Select the application session in the Application Sessions List table. Click the Shadow button to commence shadowing.

The user sees a dialog box, asking whether to allow you to shadow the session. If the user agrees, a new window appears on your screen, showing the running application. Both you and the user can control the mouse pointer and use the application.

When you have fixed the user's problem, close the shadowing window, but do not close the application. The user sees a dialog box, saying that nobody is currently shadowing the session.

The Application Sessions tab shows other application session information, such as the date and time the session started, and whether the session is suspended or currently active.

You can only shadow Windows applications and X applications. The application must not be suspended.

If the user has application sessions for two or more applications that are using shared resources, all applications that are sharing resources are displayed when you shadow the session. The button bar on the shadowing window enables you to toggle between the applications.

You can also shadow a user's session from the command line, using the `tarantella emulatorsession shadow` command.

If you are shadowing over a low bandwidth connection and have display update problems, see [“Display Update Issues When Shadowing Over a Low Bandwidth Connection” on page 220](#) for details of how to fix this.

A Kiosk Application Is Not Appearing Full-Screen

If an application that is configured to display in a kiosk window is resumed on a display that is larger or smaller than the original display, the application no longer fits the screen exactly.

The solution is to ensure that SGD scales the kiosk window to fit the screen.

In the Administration Console, go to the Presentation tab for the application object and set the Window Size to Scale to Fit Window.

Alternatively, use the following command:

```
$ tarantella object edit --name obj --scalable true
```

An Application's Animation Appears 'Jumpy'

Changing an application object's performance settings can improve the display of animation effects in the application session.

In the Administration Console, go to the Performance tab for the application object and set the Command Execution attribute to In Order. Deselect the Delayed Updates check box.

Alternatively, use the following command:

```
$ tarantella object edit --name obj \  
--execution inorder --delayed false
```

Font Problems with X Applications

If users see font problems with X applications, check the following:

Is the font size wrong?

In the Administration Console, go to the Client Device tab for the X application object and check the value of the Monitor Resolution attribute. Display the Protocol Engines → X tab for each SGD server in the array and check the value of the Monitor Resolution attribute.

The Monitor Resolution attributes are used to specify the monitor resolution, in dots per inch, that SGD reports to X applications that ask for this information. Some X applications need this value to determine what font size to use.

The default resolution can cause the X application to choose a font size larger than it normally chooses. If this happens, try reducing the resolution by specifying a smaller value, for example, 75.

Are the wrong fonts displayed?

In the Administration Console, go to the Protocol Engines → X tab for each SGD server in the array and check that the Font Path attribute is correct.

Some [X Fonts](#) are supplied with SGD. You can also configure your own X fonts. See [“How to Configure SGD to Use Your Own X Fonts” on page 157](#).

Display Problems With High Color X Applications

Several problems can occur when displaying high color X applications:

- [The X Application Fails With a Color Planes Error](#)
- [The Colors Appear Strange](#)
- [The X Application Uses Too Much Bandwidth](#)
- [8-bit Applications Exit With a PseudoColor Visual Error](#)

The X Application Fails With a Color Planes Error

If an X application fails to run and exits with errors such as “Cannot Allocate Enough Color Planes”, the application probably only displays 8-bit color. Check the display specification of the application and adjust the color depth of the application object.

In the Administration Console, go to the Presentation tab for the application object and set the Color Depth to 8-bit - 256 colors.

Alternatively, use the following command:

```
$ tarantella object edit --name obj --depth 8
```


The Colors Appear Strange

If there are any problems with appearance in 16-bit or 24-bit color applications, change the color quality of the application object.

In the Administration Console, go to the Performance tab for the application object and set the Color Quality to 16-bit, for 16-bit applications, or 24-bit, for 24-bit applications.

Alternatively, use the following command:

```
$ tarantella object edit --name obj --quality 16 | 24
```

The X Application Uses Too Much Bandwidth

If bandwidth is critical, try reducing the color quality of the application object.

In the Administration Console, go to the Performance tab for the X application object and set the Color Quality to 9-bit, or 6-bit.

Alternatively, use the following command:

```
$ tarantella object edit --name obj --quality 9 | 6
```

Note – There is no absolute guarantee of a bandwidth saving when you make this configuration change. Also, the appearance of the application might be affected adversely.

8-bit Applications Exit With a PseudoColor Visual Error

If you run an 8-bit application within a 16-bit or 24-bit high color X application session, for example from a CDE desktop, you might find the application exits with an error such as "Cannot find a matching 8-bit PseudoColor visual".

To fix this, change the color depth of the X application so that it supports multiple color depths.

In the Administration Console, go to the Presentation tab for the X application object and set the Color Depth to 16/8-bit - Thousands of Colors, or 24/8-bit - Millions of Colors.

If the 8-bit application requires the primary color depth to be 8-bit, set the Color Depth to 8/16-bit - Thousands of Colors, or 8/24-bit - Millions of Colors.

Alternatively, use the following command:

```
$ tarantella object edit --name obj --depth 16/8 | 24/8
```

Note – There are memory and performance effects of using these settings.

If the application still exits after changing the color depth, a workaround is to create a separate X application object for the application and set the color depth to 8-bit.

Clipped Windows With Client Window Management Applications

If users see clipped windows when using X applications that are configured to use Client Window Management, this means that users have displays with a greater resolution than expected.

The solution is to increase display resolution for the X Protocol Engine.

In the Administration Console, go to the Protocol Engines → X tab for each SGD server in the array and change the Client Window Size settings. In the Maximum Height and Maximum Width fields, type the highest display resolution you expect to support.

Note – Increasing the Maximum Width and Maximum Height attributes increases the memory requirements for Client Window Management applications on both client devices and SGD servers.

Alternatively, use the following command:

```
$ tarantella config edit --array \  
--xpe-cwm-maxwidth pixels \  
--xpe-cwm-maxwidth pixels
```

Emulating a Sun Keyboard

Some applications accept input from the left-hand keypad on a Sun workstation. To emulate these keys using Shift-Function key strokes on the client device, you must use a custom keymap file.

Log in to SGD as root (superuser) and make a copy of the `xuniversal.txt` keymap file. This file is located in the `/opt/tarantella/etc/data/keymaps` directory on the SGD server. Rename the file to `xsunkey.txt`.

Edit the Function key definitions in the `xsunkey.txt` file, as follows:

```
112 F1 Cancel NoSymbol NoSymbol 0x3b
113 F2 Redo NoSymbol NoSymbol 0x3c
114 F3 0x1005ff70 NoSymbol NoSymbol 0x3d
115 F4 Undo NoSymbol NoSymbol 0x3e
116 F5 0x1005ff71 NoSymbol NoSymbol 0x3f
117 F6 0x1005ff72 NoSymbol NoSymbol 0x40
118 F7 0x1005ff73 NoSymbol NoSymbol 0x41
119 F8 0x1005ff74 NoSymbol NoSymbol 0x42
120 F9 Find NoSymbol NoSymbol 0x43
121 F10 0x1005ff75 NoSymbol NoSymbol 0x44
122 F11 Help NoSymbol NoSymbol 0x57
```

This maps the client device Function keys to Sun workstation keys, as shown in the following table.

| Function Key | Sun Workstation Key |
|--------------|---------------------|
| Shift-F1 | Stop |
| Shift-F2 | Again |
| Shift-F3 | Props |
| Shift-F4 | Undo |
| Shift-F5 | Front |
| Shift-F6 | Copy |
| Shift-F7 | Open |
| Shift-F8 | Paste |
| Shift-F9 | Find |
| Shift-F10 | Cut |
| Shift-F11 | Help |

On the application server that runs the application, add the following line to the `/usr/dt/lib/bindings/xmbind.alias` file:

```
"Sun Microsystems, Inc."          sun
```

In the Administration Console, go to the Client Device tab for the user profile object. Select the Custom Value option for the Keyboard Map attribute and type `xsunkey.txt` in the field.

Alternatively, use the following command:

```
$ tarantella object edit --name obj --keymap xsunkey.txt
```

Note – The new keyboard map is used for all graphical applications for the specified user.

Display Update Issues When Shadowing Over a Low Bandwidth Connection

Display update issues might be seen when shadowing a user who has a low bandwidth connection to SGD.

The solution is to increase the X Protocol Engine queue length and optimize command execution, as follows:

- In the Administration Console, go to the Protocol Engines → X tab for each SGD server in the array and type `-mql 8192` in the Command-Line Arguments field. Alternatively, use the following command:

```
$ tarantella config edit --xpe-args "-mql 8192"
```

The changes made take effect for new X Protocol Engines only. Existing X Protocol Engines are not affected.

- In the Administration Console, go to the Performance tab for the shadowed application and set the Command Execution attribute to Optimized.

Alternatively, use the following command:

```
$ tarantella config edit --name obj --execution optimized
```

The changes made take effect when you next start the shadowed application.

Troubleshooting Mouse Drag Delay Issues

Mouse drag delay issues can result in a poor user experience when using drawing applications.

The solution is to reduce the mouse drag delay setting for the SGD Client. Add a new `<mousethrottledelaywithbutton>` entry in the `<localsettings>` section of the user's client profile. If the `<localsettings>` section is not present in the client profile, create a new section.

For example, to set the mouse drag delay to 10 milliseconds, type the following:

```
<localsettings>
  <mousethrottledelaywithbutton>10</mousethrottledelaywithbutton>
  ...
</localsettings>
```

The default value for mouse drag delay is 100 milliseconds.

Changes to client profiles only take effect for new user sessions.

Incorrect Time Zone Name Shown in Windows Applications

When time zone redirection is enabled on a Windows application server, the time zone name shown in Windows applications can sometimes be incorrect. The issue is seen when displaying Windows applications on UNIX platform client devices.

The solution is to manually set the `$TZ` time zone environment variable on the client device to the correct value for the user's location. The `tzselect` command can be used to list the possible time zone values for a geographical location.

See [“Time Zone Redirection” on page 143](#) for more details about using time zone redirection with Windows application servers.

Client Device Support

This chapter describes how to enable support for peripherals and other client device features from applications displayed in Oracle Secure Global Desktop (SGD).

This chapter includes the following topics:

- [“Printing” on page 223](#)
- [“Client Drive Mapping” on page 258](#)
- [“Audio” on page 276](#)
- [“Copy and Paste” on page 287](#)
- [“Smart Cards” on page 292](#)
- [“Serial Ports” on page 298](#)

Printing

This section describes how to configure printing services in SGD and includes the following topics:

- [“Overview of SGD Printing” on page 224](#)
- [“Setting Up Printing” on page 225](#)
- [“Configuring Microsoft Windows Application Servers for Printing” on page 225](#)
- [“Configuring UNIX and Linux Platform Application Servers for Printing” on page 228](#)
- [“Configuring an SGD Server for Printing” on page 233](#)
- [“Configuring Printing to Microsoft Windows Client Devices” on page 237](#)
- [“Configuring Printing to UNIX, Linux, and Mac OS X Platform Client Devices” on page 241](#)
- [“Managing Printing” on page 244](#)

- “Users Cannot Print From Applications Displayed Through SGD” on page 246
- “Troubleshooting Other Printing Problems” on page 254

Overview of SGD Printing

SGD supports two types of printing: PDF printing and Printer-Direct printing.

With *PDF printing*, users print from an application using an SGD PDF printer. The print job must be in PostScript™ format. The PostScript print job is sent from the application server to an SGD server, where it is converted into a Portable Document Format (PDF) file. The SGD server then sends the PDF file to a PDF viewer on the user’s client device, where the file can be viewed, saved, and printed.

With *Printer-Direct printing*, users print from an application to a printer attached to their client device. SGD does this by cooperating with the lp or lpr printing system on the SGD host and the native printing system on the application server. The print job is sent from the application server to an SGD server. The SGD server then sends the print job to the SGD Client, which sends it to the user’s client printer. If the format of the print job used by the application server is different to the format needed by the client printer, SGD converts the print job before sending it to the SGD Client.

PDF printing is usually more reliable and produces better results than Printer-Direct printing.

SGD has two PDF printers: *Universal PDF Printer* and *Universal PDF Viewer*.

On Microsoft Windows client devices, the Universal PDF Printer displays the print job as a PDF file in the Adobe® Reader, which then prints the PDF file to the user’s default printer. The Universal PDF Viewer displays the print job as a PDF file in the Adobe Reader, which the user can then decide whether to print or save.

On UNIX®, Linux, and Mac OS X platform client devices, there is no difference between the Universal PDF Printer and Universal PDF Viewer, as the print job is always displayed as a PDF file in a PDF viewer. The user can then decide whether to print or save the PDF file.

SGD uses *distributed printing*. Print jobs are sent to the SGD server hosting the user’s application session. This means that a user’s print jobs are distributed across the array, and there are no bottlenecks or single points of failure.

SGD supports printer-direct printing to PostScript, Printer Command Language (PCL), and text-only printers attached to the user’s client device. The SGD `tta_print_converter` script performs any conversion needed to format print jobs correctly for the client printer. To convert from Postscript to PCL, Ghostscript must be installed on the SGD server.

Setting Up Printing

Setting up printing involves the following configuration steps:

1. Configure application server for printing.
The required configuration for the application server depends on the application server platform.
See [“Configuring Microsoft Windows Application Servers for Printing”](#) on page 225.
See [“Configuring UNIX and Linux Platform Application Servers for Printing”](#) on page 228.
2. Configure the SGD servers for printing.
See [“Configuring an SGD Server for Printing”](#) on page 233.
3. Configure printing to client devices.
The required configuration depends on the client device platform.
See [“Configuring Printing to Microsoft Windows Client Devices”](#) on page 237.
See [“Configuring Printing to UNIX, Linux, and Mac OS X Platform Client Devices”](#) on page 241.

Configuring Microsoft Windows Application Servers for Printing

The configuration required to print from applications running on a Microsoft Windows application server depends on the version of the Microsoft Remote Desktop Protocol (RDP) protocol that is used to connect to the application server. See the following:

- [“Configuring Printing for Microsoft RDP 5.0 or Later”](#) on page 225
- [“Configuring Other Microsoft Windows Application Servers for Printing”](#) on page 228

Configuring Printing for Microsoft RDP 5.0 or Later

SGD automatically creates printer queues in the Windows application session if the application server supports Microsoft RDP version 5.0 or later. This applies to Microsoft Windows 2000 Server and later application servers.

When a user starts or resumes a Windows application, the SGD Client sends information about the client's printers to SGD. SGD supplies this information to the application server and the application server then creates, or *maps*, the printers in the Windows Terminal Services session. The user sees the printers that are attached to the client device and also the printers that are attached directly to the application server.

To be able to create a client printer in the Microsoft Windows application session, the following must be true:

- Printer mapping must be enabled on the application server, see [“Configuring Microsoft Windows Terminal Services for Use With SGD” on page 140](#) for details.
- The SGD Client must determine the name of the printer driver for the client printer and send it to application server.
- The printer driver for the client printer must be installed on the application server.

The printer drivers that must be installed on the application server are as follows:

- **PDF printing** – The printer drivers selected for use with PDF printing.

See [“Configuring the Printers Available in Windows Terminal Services Sessions” on page 226](#) for information on selecting printer drivers.

- **Printer-Direct printing** – The printer driver for every client printer.

For Microsoft Windows client devices, you can use printer driver mapping to map one printer driver name to another. See [“Printer Driver Mapping” on page 238](#).

For UNIX, Linux, and Mac OS X platform client devices, the printer configuration files specify the printer driver that is used. See [“Configuring Printing to UNIX, Linux, and Mac OS X Platform Client Devices” on page 241](#).

SGD Administrators can control the SGD printers that are available in Windows Terminal Services sessions. See [“Configuring the Printers Available in Windows Terminal Services Sessions” on page 226](#).

Configuring the Printers Available in Windows Terminal Services Sessions

SGD enables Administrators to control the printers that are available in Windows Terminal Services sessions. You can configure the printers, as follows:

- **Globally.** In the Administration Console, go to the Global Settings → Printing tab.
- **Individually.** In the Administration Console, go to the Printing tab for an organization, an organizational unit, a user profile, or a Windows application object.

If you configure an organization or organizational unit object, this affects all the users in that organization or organizational unit.

If you configure a Windows application object, this overrides the printing configuration for organization, organizational unit, or user profile objects. The order of precedence for printing configuration is: Windows application → user profile → organizational unit → organization.

You can set the following attributes on the Printing tab.

TABLE 5-1 Attributes Used to Configure Printing From Terminal Services Sessions

| Attribute | Description |
|--|--|
| Client Printing | Controls the client printers users can print to, either all client printers, the default client printer, or no client printers. By default, users can print to all their client printers. |
| Universal PDF Printer | Enables the Universal PDF Printer printer. |
| Make Universal PDF Printer the Default | Sets the Universal PDF Printer printer as the client device's default printer for Windows applications. |
| Universal PDF Viewer | Enables the Universal PDF Viewer printer. |
| Make Universal PDF Viewer the Default | Sets the Universal PDF Viewer printer as the client device's default printer for Windows applications. |
| Postscript Printer Driver | The name of the PostScript printer driver to use for PDF printing. |

Note – Any configuration changes you make on the Printing tab only take effect for new user sessions.

If you make a PDF printer the default printer for Windows applications and SGD is configured to only allow users to print to their default printer, users see two printers in their Windows application session. The user's default client printer and the PDF printer are shown.

To use PDF printing, you must install the PostScript printer drivers you want to use for PDF printing on the application server. Make sure the printer drivers have sufficient features for your users. By default, SGD is configured to use the HP Color LaserJet 2800 Series PS printer driver. The printer driver name entered in the Postscript Printer Driver field on the Printing tab must match the name of the printer driver installed on the application server *exactly*. Pay particular attention to the use of capitals and spaces. The

`/opt/tarantella/etc/data/default.printerinfo.txt` file contains all the common printer driver names, ordered by manufacturer. To avoid errors, copy and paste the driver name from this file.

Note – If a PDF viewer is not configured on the client device, the PDF printers are not available in the Windows application session, even if a PDF printer is enabled.

Configuring Other Microsoft Windows Application Servers for Printing

To print from an Microsoft Windows application server that uses a version of the Microsoft RDP protocol earlier than version 5.0, you must configure a Line Printer Remote (LPR)-compatible Transmission Control Protocol/Internet Protocol (TCP/IP) printer on the application server. Configure the printer to send print jobs to the primary SGD server in the array. Consult your system documentation for details of how to configure printers.

Note the following limitations:

- **PDF printing is not supported.**
- **No multiple printer support.** You can only print to the client device's default printer. It is not possible for users to select a printer. If a user needs to print to a different printer, they have to log out of SGD, change their default printer, and then log in again.
- **Print jobs might be deleted.** When a print job is transferred from the application server to an SGD server, the user's SGD name is needed to identify which client device to send the print job to. With some versions of Microsoft Windows, there is no direct way to associate print jobs with SGD users. If SGD cannot identify which user has printed a particular job, the print job is deleted. This might happen, for example, if two users log in to the application server with the same name.
- **Distributed printing is not available.** All print jobs are directed through the primary server in an SGD array.

Configuring UNIX and Linux Platform Application Servers for Printing

To use PDF printing from a UNIX or Linux platform application server, you must install at least one SGD printer queue on the application server. You do not have to install printer queues for the Universal PDF Printer and Universal PDF Viewer. However, if a UNIX or Linux platform application you are using does not allow you to configure printer arguments, or does not allow you to specify the Universal PDF Printer and Universal PDF Viewer because their names contain spaces, you must install an additional printer queue called `tta_pdfprinter` and print to that queue.

To use Printer-Direct printing from a UNIX or Linux platform application server, you must install SGD printer queues as follows:

- **Single printer queue.** Install an SGD printer queue for the primary SGD server in the array. All print jobs are directed to the primary SGD server, and the primary server sends the print jobs to the client device.
- **Multiple printer queues.** Install an SGD printer queue for each SGD server in the array. Each printer queue redirects print jobs to an SGD server, and the SGD server sends the print jobs to the client device.

Note – It is best to use multiple printer queues so that print jobs are distributed across the array, and there are no bottlenecks or single points of failure

You configure printer queues with the SGD printer queue installation script. See [“How to Install an SGD Printer Queue on a UNIX or Linux Platform Application Server”](#) on page 229.

The SGD printer queue installation scripts installs replacement `lp` or `lpr` scripts. These are used instead of the standard scripts, to ensure that print jobs contain enough information for SGD to be able to identify the user that printed them. See [“Printing With the SGD `lp` and `lpr` Scripts”](#) on page 232 for details.

▼ How to Install an SGD Printer Queue on a UNIX or Linux Platform Application Server

If the application server is also an SGD server, a printer queue is installed automatically when you install SGD.

1. **Copy the `/opt/tarantella/bin/scripts/prtinstall.en.sh` script from an SGD server to a temporary directory on the application server.**
2. **Log in to the application server as superuser (root).**
3. **Change to the temporary directory.**
4. **Run the script to install the printer queue.**

See [“The SGD Printer Queue Installation Script”](#) on page 230 for details of all the command options for the SGD printer queue installation script.

- If the array consists of a single SGD server, use the following command:

```
# sh prtinstall.en.sh
```

When prompted, type the full Domain Name System (DNS) name of the SGD server.

- If the array contains more than one SGD server, create a printer queue for each SGD server in the array. Use the following command:

```
# sh prtinstall.en.sh --ttahost DNS-name --appprinter name
```

The *DNS-name* is the full DNS name of an SGD server. The name of each printer queue, as specified by the `--appprinter` argument, can be anything you like but it must be unique.

If you use the Common UNIX Printing System (CUPS), you might have to use the `--cups` option with `prtinstall.en.sh`, to indicate that you are using CUPS. You might also have to reconfigure CUPS. See [“Configuring Printing for CUPS” on page 232](#).

The SGD Printer Queue Installation Script

The SGD printer queue installation script, `prtinstall.en.sh`, installs an SGD printer queue on a UNIX or Linux platform application server. It also installs the SGD replacement `lp` or `lpr` scripts.

The `prtinstall.en.sh` script is located in the `/opt/tarantella/bin/scripts` directory on the SGD server.

You must be superuser (root) to run this script.

The syntax for the script is as follows:

```
sh prtinstall.en.sh [--ttahost SGD_hostname]  
                    [--ttapprinter printer_name]  
                    [--appprinter printer_name]  
                    [--uninstall [printer_name]]  
                    [--cups y | n | auto]  
                    [--cupsconf filename]  
                    [--cupscontrol filename]  
                    [--gsbindir gs_bin_dir]  
                    [--append]  
                    [--help]
```

The following table describes the available options for the script.

| Option | Description |
|---|---|
| <code>--ttahost SGD_hostname</code> | Fully qualified DNS name of an SGD server |
| <code>--ttaprinter printer_name</code> | Use this option to specify a name for the printer queue. Use this if the SGD server is also used as an application server. If you do not use this option, the printer is created with the default name of <code>tta_printer</code> . |
| <code>--appprinter printer_name</code> | Use this option to specify a name for the printer queue on a UNIX or Linux platform application server. If you do not use this option, the printer queue is created with the default name of <code>tta_printer</code> . |
| <code>--uninstall [printer_name]</code> | Uninstalls an SGD printer queue. If you do not specify a printer queue, you are prompted for one. |
| <code>--cups y n auto</code> | Indicates that you are using CUPS. If you do not use this option, a default of <code>auto</code> is assumed and this means SGD automatically detects whether CUPS is being used. If CUPS is incorrectly detected, use this option to tell SGD whether CUPS is being used (<code>y</code>) or not (<code>n</code>). |
| <code>--cupsconf filename</code> | Specifies the path to the CUPS configuration file. If you do not use this option, the CUPS configuration file is assumed to be <code>/etc/cups/cupsd.conf</code> . |
| <code>--cupscontrol filename</code> | Specifies the path to the CUPS startup script. If you do not use this option, the CUPS startup script is assumed to be <code>/etc/init.d/cups</code> . |
| <code>--gsbindir gs_bin_dir</code> | Use this option to specify the directory where Ghostscript is installed. Use this option if Ghostscript is not installed in one of the default locations, or to specify the version of Ghostscript to use, if more than one version is installed. Only use this option if you are running the printer queue installation script on the SGD host. See “Checking the Ghostscript Installation on the SGD Host” on page 233 for details. |
| <code>--append</code> | Installs an additional printer queue rather than replacing the existing printer queue(s). |
| <code>--help</code> | Shows the list of <code>prtinstall.en.sh</code> script options. |

The following example installs an SGD printer called `tta_london` on an application server.

```
# sh prtinstall.en.sh --appprinter tta_london
```

Configuring Printing for CUPS

SGD printing only works with CUPS version 1.1.19 or later. The following configuration changes might be needed to enable printing with CUPS:

- **CUPS LPD compatibility mode must be enabled for any LPD clients.**

If you have any Line Printer Daemon (LPD) clients on your application server, you must enable the CUPS LPD compatibility mode so that CUPS can accept remote print jobs from LPD clients. The [CUPS Software Administrators Manual](#) explains how you enable LPD compatibility mode.

- **CUPS raw printing must be enabled.**

On the host where SGD is installed, enable raw printing in CUPS, by editing the `/etc/cups/mime.convs` and `/etc/cups/mime.types` files. These files contain comments explaining how to do this. Search for comments containing the word “raw”.

Note – After making changes to your CUPS configuration, you might have to restart the CUPS daemon.

To use CUPS for printing, you must use the `/opt/tarantella/bin/lp` script.

Printing With the SGD `lp` and `lpr` Scripts

The SGD printer queue installation script, `prtinstall.en.sh`, installs the SGD `lp` or `lpr` replacement scripts. Users must use these replacement scripts when they print from a UNIX or Linux platform application server. The replacement scripts ensure that print jobs contain enough information for SGD to be able to identify the user that printed them.

The SGD login scripts set the user’s `PATH` to ensure that the replacement scripts take precedence over the system scripts. However, if the application uses a full path name, for example `/usr/bin/lp`, or modifies `PATH` itself, you must reconfigure the application to use `/opt/tarantella/bin/lp` or `/opt/tarantella/bin/lpr`.

Users print with the replacement scripts as follows:

```
$ lp -d printer file
```

```
$ lpr -P printer file
```

If the `-d` or `-P` argument is omitted, the output goes to the client’s default printer. How you specify the *printer* depends on the client device. See “[Configuring Printing to Microsoft Windows Client Devices](#)” on page 237 and “[Configuring Printing to UNIX, Linux, and Mac OS X Platform Client Devices](#)” on page 241 for details.

Configuring an SGD Server for Printing

Configuring an SGD server for printing involves the following configuration steps:

- Ghostscript might need to be installed on the SGD host. SGD might need to be configured to find the Ghostscript installation.
See [“Checking the Ghostscript Installation on the SGD Host” on page 233](#).
- The SGD host might need to be configured to accept remote print requests.
See [“Configuring the SGD Host to Accept Remote Print Requests” on page 234](#).
- SGD might need to be configured to convert print jobs between different formats.
See [“Configuring SGD Print Job Conversion” on page 235](#).

Checking the Ghostscript Installation on the SGD Host

SGD uses [Ghostscript](#) to convert print jobs into PDF files. To use PDF printing, Ghostscript version 6.52 or later must be installed on the SGD host. Your Ghostscript distribution must include the `ps2pdf` program.

With Printer-Direct printing, the `tta_print_converter` script uses Ghostscript to convert print jobs from PostScript to PCL format. For best results, download and install the additional fonts.

Ghostscript is not included with the SGD software.

When you install SGD, it automatically detects Ghostscript if it is installed in one of the following locations:

- `/usr/local/bin`
- `/usr/bin`
- `/usr/sfw/bin`
- `/opt/sfw/bin`
- `/bin`
- `/usr/sbin`
- `/sbin`
- `/usr/lbin`

If Ghostscript is installed in a different location, run the SGD printer queue installation script on the SGD host. Use the `--gsbindir` option of the script to configure the location of Ghostscript. See [“The SGD Printer Queue Installation Script” on page 230](#) for more details.

If more than one version of Ghostscript is installed, run the SGD printer queue installation script with the `--gsbindir` option, to tell SGD which version to use.

If Ghostscript is not installed on the SGD host, or your Ghostscript distribution does not include the `ps2pdf` program, install Ghostscript and then run the SGD printer queue installation script.

Using the `gstest` Script to Test a Ghostscript Installation

You can use the `gstest` script to test the Ghostscript installation on an SGD host. This script is run by default when you install SGD.

The `gstest` script checks for errors in the Ghostscript installation and uses `ps2pdf` to generate a test PDF file. Script output is reported on-screen, and is also written to the `/opt/tarantella/var/log/print.log` file.

You run `gstest` as follows:

```
# /opt/tarantella/bin/scripts/gstest
```

Using `gstest` in this way performs a basic test of the font installation on the SGD host and generates a fonts test file, `/opt/tarantella/var/info/sample.pdf`. If Ghostscript fonts are installed correctly, the `sample.pdf` file contains three lines, each rendered in a different font. The fonts used are listed in the `/opt/tarantella/var/log/print.log` file.

Alternatively, you can specify an input file and output file to use with `gstest`. For example:

```
# cd /opt/tarantella/bin/scripts
# gstest /tmp/myPostScriptFile.ps /home/indigojones/myPDFFile.pdf
```

If you do not specify an output file, `gstest` creates an output PDF file at `/tmp/sgd_sample.pdf`.

Note – If you specify your own input file, `gstest` does not generate the fonts test PDF file, `/opt/tarantella/var/info/sample.pdf`.

Configuring the SGD Host to Accept Remote Print Requests

Print jobs are sent from the application server to an SGD server, and then from the SGD server to the client device. To be able to direct print jobs from an application server to a client device, the SGD host must be configured to accept remote print requests. How you do this varies for each platform. Check your System Administration documentation for information about this.

For example, if you are using `lpd` on Linux systems, you must add an entry in the `/etc/hosts.equiv` or `/etc/hosts.lpd` file, if available, for each application server that might send a print request. After making these changes, remember to restart the `lpd` daemon.

Configuring SGD Print Job Conversion

With Printer-Direct printing, print jobs are sent from an application server to an SGD server. The SGD server then sends the print job to the client device, which sends it to the user's printer. When print jobs arrive at the SGD server, they might need to be converted to a format suitable for the client printer.

Note – Print jobs from Windows RDP sessions are never converted, because they are assumed to be in the correct format.

To decide whether a print job needs to be converted, the SGD server checks a printer type configuration file to see whether the format used by the client printer matches the format used by the application server. If the format matches, the print job is forwarded to the client device printer without any conversion. If the formats do not match, the SGD server converts the print job to the right format using the `tta_print_converter` script.

To ensure that print jobs are formatted correctly, you might have to edit a printer type configuration file and the `tta_print_converter` script. This is described in the following sections.



Caution – Only edit these files if you have to use Printer-Direct printing and need to resolve issues with print job formats. In most cases, PDF printing provides a better solution for issues with print job formats.

Printer Type Configuration Files

SGD uses the following configuration files to determine the printer type:

- **Microsoft Windows client devices.** The `/opt/tarantella/etc/data/printertypes.txt` file is used.
See [“Configuring Printing to Microsoft Windows Client Devices” on page 237](#).
- **UNIX, Linux, and Mac OS X platform client devices.** One of the following files is used:
 - `/opt/tarantella/etc/data/default.printerinfo.txt` – This is the global configuration file.

- `$HOME/.tarantella/printerinfo.txt` – This is a user-specific configuration file.

See [“Configuring Printing to UNIX, Linux, and Mac OS X Platform Client Devices”](#) on page 241.

You can edit these files if you want to support particular printers, or to add new types of printer.

Note – If you add a new printer type, you might also have to edit the `tta_print_converter` script.

If there is insufficient detail or inaccurate mappings in these files, SGD might convert print jobs unnecessarily, or not at all.

The tta_print_converter Script

The `tta_print_converter` script converts print jobs from the format used by the application server to the format required by the client device, as determined by the printer type. By default, the script recognizes PostScript and non-PostScript formats. To convert print jobs from PostScript to PCL, Ghostscript must be installed on the SGD host. See [“Checking the Ghostscript Installation on the SGD Host”](#) on page 233 for more information about installing and configuring Ghostscript for SGD printing.

You can edit the `tta_print_converter` script to recognize and convert between different print job formats, or to add support for a new printer type.

Note – You must log on as superuser (root) to edit the script.

The `tta_print_converter` script is in the `/opt/tarantella/bin/scripts` directory. The script includes comments, to help you to customize it.

The shell function `GetDataType` determines the print job format from the first 128 bytes of the print job. The data is Uniform Resource Locator (URL)-encoded, for example, the `%` character is encoded as `%25`.

The client printer type is passed to this script in upper case, for example, `POSTSCRIPT` or `MYNEWTTYPE`.

If you experience problems printing to a PCL printer, the `tta_print_converter` script contains some code which has been commented out. You can use this code to see if this solves the problem.

Configuring Printing to Microsoft Windows Client Devices

The configuration required for printing to Microsoft Windows client devices depends on whether you are using PDF printing or Printer-Direct printing, as described in the following sections.

PDF Printing

To be able to use PDF printing, the [Adobe Reader](#) version 4.0 or later must be installed on the client device.

From a Microsoft Windows application, you print in the normal way, and select either the Universal PDF Printer or the Universal PDF Viewer in the application's Print dialog.

From an application running on a UNIX or Linux platform application server, you print in the normal way, using the SGD replacement `lp` or `lpr` scripts. You select a PDF printer as part of the print command, for example:

```
$ /opt/tarantella/bin/lp -d "Universal PDF Printer" filename
```

```
$ /opt/tarantella/bin/lpr -P "Universal PDF Viewer" filename
```

Note – The *filename* must be a PostScript file, so the application must be able to output PostScript.

When users print, the PDF file is displayed in the Adobe Reader. If the Universal PDF Printer is selected, the PDF file is printed automatically to the user's default printer. The Adobe Reader runs minimized and does not exit when the print job has finished. If the Universal PDF Viewer is selected, the PDF file is displayed in the Adobe Reader window. The user can then decide whether to print or save the file.

On UNIX, Linux, and Mac OS X platform client devices, the PDF file is displayed either in the default PDF viewer or in the PDF viewer configured in the client profile. The user can then decide whether to print or save the PDF file. There is no difference between the Universal PDF Printer and the Universal PDF Viewer, as the print job is always displayed in a PDF viewer.

Printer-Direct Printing

This section describes the configuration that might be needed when using Printer-Direct printing to print to Microsoft Windows client devices and includes the following topics:

- [“Printer Driver Mapping” on page 238](#)
- [“The Printer Types Configuration File” on page 239](#)
- [“Printing From a UNIX or Linux Platform Application Server” on page 240](#)

Printer Driver Mapping

When printing from a Microsoft Windows application, the large number and variety of client printers available can cause problems. The majority of the problems are caused by not having the correct printer drivers installed on the application server. One solution is to use PDF printing. Another solution, for Windows client devices only, is to use printer driver mapping.

Printer driver mapping enables you to map one printer driver name to another. You do this by editing the [Previous Names] section of the `/opt/tarantella/etc/data/default.printerinfo.txt` file.

The following is an example of an entry in a `default.printerinfo.txt` file:

```
[Previous Names]
"HP LaserJet 5" = "my HP driver", "my other HP driver"
```

This means that if users have client printers that use either the "my HP driver" or "my other HP driver" printer driver, SGD uses the "HP LaserJet 5" printer driver when creating the printer.

You can also use wild-card characters, such as * and ?, on the right hand side of the = sign. Use * to mean any string of characters, including an empty string and ? to mean any single character. This is useful, for example, to create generic printer mappings where you have a wide variety of client devices.

For example, if the file contains the following entry:

```
[Previous Names]
"HP LaserJet 5" = "hp*laserjet 5*"
```

All printer driver names like "HP LaserJet 5", "HP LaserJet 5M", and "HP Color LaserJet 5" are mapped to the printer driver "HP LaserJet 5".

The `default.printerinfo.txt` file contains more detailed instructions on how to create the mappings.

The Printer Types Configuration File

For Microsoft Windows client devices, SGD uses the `/opt/tarantella/etc/data/printertypes.txt` file to determine whether to convert a print job from one format to another before sending it to the client device. The `printertypes.txt` file maps printer drivers, for example, `pscript.dll`, to printer types, for example PostScript.

Note – Print jobs from Windows RDP sessions are never converted, because they are assumed to be in the correct format.

The `printertypes.txt` file includes comments to help you to customize it. By default, the file includes mappings for PostScript, PCL, and text-only printers. You must log on as superuser (root) to edit this file.

Note – The `printertypes.txt` file used for Windows clients also contains entries for UNIX platform and Apple Macintosh clients. This is used only as a fallback. For UNIX or Linux platforms, it maps UNIX types to printer types. For Apple Macintosh, it maps printer names to printer types.

To find out the name of the printer driver used by a client device, print a test page and check the Driver Name field.

To add support for a new printer type, add lines following the same pattern. For example:

```
MyNewType=mydriver.drv
```

For example, a client device, `cairo`, runs Windows 2000 and its default printer is PCL. The printer driver used is `unidrv.dll`. The `[Windows*]` section in `printertypes.txt` has the following format:

```
[Windows*]  
PostScript=pscript5.dll;pscript.dll  
PCL=rasdd.dll  
PostScript=*
```

As there is no specific match for `unidrv.dll`, the final entry applies: PostScript. This means that when the user prints, print jobs are incorrectly converted to PostScript before being sent to `cairo`.

To fix this, edit `printertypes.txt` as root and add a specific match for `unidrv.dll` as follows:

```
PCL=rasdd.dll;unidrv.dll
```

Following this change, SGD correctly identifies the printer configured on cairo, and print jobs are converted to PCL for that client device.

Printing From a UNIX or Linux Platform Application Server

When printing from a UNIX or Linux platform application server to a Microsoft Windows client device, users can specify the printer they print to by using any of the following:

- The Universal Naming Convention (UNC) name of a network printer accessible to the client, for example:

```
$ lp -d '\\\\PRTSERVER\\HPLJ5' filename
```

- A “friendly” name, for example:

```
$ lpr -P label-printer filename
```

- A port on the client, for example:

```
$ lpr -P LPT1: filename
```

To use a UNC name, you must enclose the printer name in quotes and escape every backslash with an extra backslash, as shown in the previous example. As different shells process backslashes differently, you might need to experiment with the number of backslashes. You can also use underscores instead of backslashes, for example:

```
$ lp -d __PRTSERVER_HPLJ5 filename
```

Note – Using underscores only works if the first two characters of the printer name are underscores.

You can avoid problems with UNC names by using a “friendly” name. You configure “friendly” names in the `/opt/tarantella/etc/data/printernamemap.txt` file. The entries in this file map “friendly” names to UNC names, for example:

```
"label-printer"="\\PRTSERVER\\HPLJ5"
```

Note – You do not have to escape any backslashes.

Configuring Printing to UNIX, Linux, and Mac OS X Platform Client Devices

The configuration required for printing to UNIX, Linux, and Mac OS X platform client devices depends on whether you are using PDF printing or Printer-Direct printing, as described in the following sections.

PDF Printing

To be able to use PDF printing, a PDF viewer must be installed on the client device. SGD supports the following PDF viewers by default.

| Client Platform | Default PDF Viewer |
|-------------------------------|---|
| Solaris OS on SPARC platforms | Adobe Reader (acroread) GNOME PDF Viewer (gpdf) |
| Solaris OS on x86 platforms | GNOME PDF Viewer (gpdf) |
| Linux | GNOME PDF Viewer (gpdf) Evince Document Viewer (evince) X PDF Reader (xpdf) |
| Mac OS X | Preview App (/Applications/Preview.app) |

Note – The Adobe Reader PDF viewer must support the `-openInNewWindow` command option. The Preview App PDF viewer must support the `open -a` command option.

To be able to use a default PDF viewer, the application must be on the user’s `PATH`.

If an alternative PDF viewer is preferred, the command for the alternative viewer application can be configured in the user’s client profile. In the profile you enter either the command or the *full path* to the command, depending on whether the application is on the user’s `PATH`. See “[Client Profile Settings](#)” on page 315 for details.

From a Microsoft Windows application, you print in the normal way, and select either the Universal PDF Printer or the Universal PDF Viewer in the application’s Print dialog.

From an application running on a UNIX or Linux platform application server, you print in the normal way, using the SGD replacement `lp` or `lpr` scripts. You select a PDF printer as part of the print command, for example:

```
$ /opt/tarantella/bin/lp -d "Universal PDF Printer" filename
```

```
$ /opt/tarantella/bin/lpr -P "Universal PDF Viewer" filename
```

Note – The *filename* must be a PostScript file, so the application must be able to output PostScript.

The PDF file is displayed either in the default PDF viewer or in the PDF viewer configured in the client profile. The user can then decide whether to print or save the PDF file. There is no difference between the Universal PDF Printer and the Universal PDF Viewer, as the print job is always displayed in a PDF viewer.

Printer-Direct Printing

To use Printer-Direct printing to print to printers attached to UNIX, Linux, or Mac OS X platform client devices, the client printers must be defined in one of the following printer configuration files:

- **Global printer configuration file –**

`/opt/tarantella/etc/data/default.printerinfo.txt.`

This file sets the defaults for *all* users printing through that SGD server. As this file is not replicated across the array, you have to manually copy it to the other SGD servers.

- **User-specific printer configuration file –**

`$HOME/.tarantella/printerinfo.txt.`

The user-specific printer configuration file is optional and has to be manually created on client devices. Users can create their own file or you can use the global configuration file as a template and distribute it to users. This file contains the settings for an individual user regardless of the SGD server they print through. The settings in this file take precedence over the settings in the global configuration file.

The format of the global and user-specific printer configuration file is the same:

```
[UNIX]
"printer-name" = "windows-driver" printer-type
"printer-name" = "windows-driver" printer-type
...
```

printer-name is the name of the printer as it is known to the `lp` or `lpr` system on the client. The printer name must be enclosed in straight quotation marks (") and be followed by an Equal (=) sign. This is the name that users can specify when printing from a UNIX or Linux platform application server. It is also the name that displays in the Print dialog when users print from a Microsoft Windows application server.

windows-driver is the name of the printer driver to use when printing from a Microsoft Windows application server. The printer driver name must be enclosed in double quotes. The name of the printer driver must match the name of the printer driver installed on the Windows application server *exactly*. Pay particular attention to the use of capitals and spaces. The `default.printerinfo.txt` file contains all the common printer driver names ordered by manufacturer. To avoid errors, copy and paste the driver name from this file.

printer-type is the format to be used for the print job. The values can be `PostScript`, `PCL` or `Text`. This information is optional, but if it is missing, `PostScript` is used by default. This information is used to determine whether SGD converts the print job from the format used by the application server to the format needed by the client printer. See also [“Configuring SGD Print Job Conversion” on page 235](#).

The first printer listed in the `[UNIX]` section is the client’s default printer.

When SGD is first installed, the `default.printerinfo.txt` file contains the following entry:

```
[UNIX]
"_Default" = "QMS 1060 Print System" PostScript
```

With this configuration, when users print from a Windows application server, they see a printer called `_Default`. This printer prints to the default printer on the client using a basic PostScript printer driver, “QMS 1060 Print System”.

Note – This means that a printer is available in the Windows application, even if there is no printer connected to the client device.

For example, if an SGD user’s `$HOME/.tarantella/printerinfo.txt` file contains the following entries:

```
[UNIX]
"drafts" = "HP Diskette 970Cxi" PCL
"salespersons" = "HP Lacerate 5/5M" PostScript
```

When the user prints from a Microsoft Windows application server to a UNIX platform client device, the following printers are available:

- `drafts`
- `salespersons`

The user's default printer is `drafts`, which in this example has been defined as a PCL printer.

Managing Printing

This section describes the print job management features of SGD and includes the following topics:

- [“The `tarantella print` Command” on page 244](#)
- [“Setting a Time Limit for Print Jobs” on page 245](#)
- [“User Management of Print Jobs” on page 245](#)

The `tarantella print` Command

SGD Administrators control printing services with the `tarantella print` command. This command enables you to do the following:

- List spooled print jobs and identify the SGD users they belong to. You can use this to check that print jobs from the application server printing system have reached the SGD print queue.
- Remove print jobs from the SGD print queue.
- Pause and restart SGD printing services.
- Move print jobs from one SGD server to another.

The syntax for the `tarantella print` command is as follows:

```
tarantella print start | stop | status | pause | resume | list  
| cancel | move
```

The following table shows the available subcommands for `tarantella print`.

| Subcommand | Description |
|------------|--|
| cancel | Cancels print jobs |
| list | Lists print jobs |
| move | Moves queued print jobs from one SGD server to another |
| pause | Pauses printing temporarily |
| resume | Resumes printing |
| start | Starts printing services for the array |
| status | Displays information about printing services |
| stop | Stops printing services |

Setting a Time Limit for Print Jobs

SGD Administrators can set a time limit on how long a print job can remain on an SGD server before it is deleted. This is useful if you have to manage a high volume of printing.

To specify the number of hours that print jobs remain on the server, use the following command:

```
$ tarantella config edit \  
--tarantella-config-array-printjoblifetime hours
```

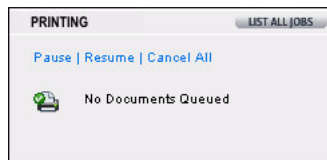
To return SGD to its default behavior, so that print jobs remain on the server indefinitely, use the following command:

```
$ tarantella config edit \  
--tarantella-config-array-printjoblifetime 0
```

User Management of Print Jobs

Users can manage their own print jobs from the Printing area on the webtop, as shown in [FIGURE 5-1](#).

FIGURE 5-1 Printing Area on the SGD Webtop



The Printing area shows the number of jobs currently in the print queue and the controls for managing print jobs.

When documents are printing, the webtop tells a user how many print jobs they have in the queue. Users can click Cancel All to delete all pending print jobs.

Users can also click Pause to temporarily stop printing. When printing is paused, any print jobs that are pending are held in a queue until the user either cancels them or resumes printing. Click Resume to start printing again. The printer icon changes to show you when printing is paused.

To manage print jobs individually, click List all jobs. The webtop displays a list of all the print jobs the user has in the queue, along with information about the job, for example the number of copies and the selected printer.

If you pause printing, click the Resume button to print just that one print job.

To cancel a print job, click the Cancel button.

When printing from a Microsoft Windows application server, or a UNIX or Linux platform application server, users can choose which printer they print to. If the user does not select a printer, the output goes to their default printer. For all other application servers, the output always goes to the client device's default printer.

Users can see which printer is their default printer by pointing with the mouse at the printer icon on their webtop. A popup displays the name of the default printer.

If a user wants to change their default printer, they must log out of SGD, change the default printer and then log in to again.

Users Cannot Print From Applications Displayed Through SGD

Use the following checklists to diagnose and fix the problem:

- [“Client Devices Checklist” on page 246](#)
- [“Application Server Checklist” on page 248](#)
- [“SGD Server Checklist” on page 250](#)

If this does not resolve the problem, follow the steps in [“Tracing a Print Job” on page 251](#).

Client Devices Checklist

Use the following client device troubleshooting steps to diagnose printing problems in SGD.

Does SGD Support Printing for the Client Device or Printer Type?

Check the Printing Area on the webtop. Does the printer icon contain a red cross and is the message “No Client Printer Available” displayed? If so, this means that SGD does not support printing for this client device or printer type, or that there was an error creating client printers.

Is Printing Paused on the Client Device?

Make sure that the user has not paused printing. Check the Printer Paused icon is not displayed.

Use the `tarantella webtopsession list` command to see whether the user has paused printing.

Is the Printer Configured Correctly?

Make sure that the printer is correctly configured, for example by printing a web page to the printer from a web browser on the client device. Depending on the application server, some print jobs can only go to the client device's default printer.

If printing to a UNIX, Linux, or Mac OS X platform client device, check that you have configured printing for these client types. See [“Configuring Printing to UNIX, Linux, and Mac OS X Platform Client Devices”](#) on page 241.

For PDF Printing, is the PDF Viewer Installed on the Client?

To be able to use PDF printing in SGD, a PDF viewer must be installed on the client device.

Check that the supported viewer, or the user's preferred viewer, is installed on the client and that the application is executable.

On UNIX, Linux, or Mac OS X system client devices, check that the user has read and write access to the /tmp directory.

If the PDF viewer is Adobe Reader (acroread), check that the viewer supports the -openInNewWindow command option. If the PDF viewer is Preview app (/Applications/preview.app), check that the viewer supports the open -a command option.

If a PDF viewer is not installed or accessible, the SGD PDF printers are available to the user.

For PDF Printing From a UNIX or Linux Platform Application Server, is the Print Job in the Right Format?

If the user's PDF viewer starts, but they receive a file format error, check that the format of the file being printed on a UNIX or Linux platform application server is PostScript.

Does the User Have the Necessary Registry Permissions?

On Microsoft Windows client devices, users must have write access to the HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Cryptography\RNG\Seed registry key and read access to the rest of the registry.

This access is required by several of the Windows application programming interfaces (APIs) for printing.

Application Server Checklist

Use the following application server troubleshooting steps to diagnose printing problems in SGD.

Is a Printer Configured on the Application Server?

Before users can print, you might need to configure a SGD printer on your application server. See the following:

- [“Configuring Microsoft Windows Application Servers for Printing” on page 225.](#)
- [“Configuring UNIX and Linux Platform Application Servers for Printing” on page 228.](#)

Is the Printer Created in the Windows Application Session?

If the user is trying to print from a Microsoft Windows application server, accessed using Windows Terminal Services, then the user’s printers are configured automatically. See [“Configuring Printing for Microsoft RDP 5.0 or Later” on page 225.](#) If not, check the System event log on the application server for the following errors:

- Event ID: 1111 Description: Driver *drivername* required for printer *printertype* is unknown. Contact the administrator to install the driver before you log in again.
- Event ID: 1105 Description: Printer security information for the *printername* / *clientcomputername* /Session *number* could not be set
- Event ID: 1106 Description: The printer could not be installed.

These errors indicate that the printer driver for the client printer might not be supported by the application server. Either install the printer driver on the application server, or see [“Printer Driver Mapping” on page 238](#) for details of how to support other printer drivers, including using wildcards to support a wide range of printer driver names.

It is also worth checking that the name of the printer driver in `/opt/tarantella/etc/data/default.printerinfo.txt`, or the user’s `$HOME/.tarantella/printerinfo.txt`, matches the name of the driver on the application server.

If this does not resolve the problem, see Microsoft Knowledge Base article 239088 for more details.

Is the Application Printing to the Correct Printer?

The application must print to the printer queue you have configured. On UNIX or Linux platform application servers, the `prtinstall.en.sh` script creates a printer queue named `tta_printer` by default.

On UNIX or Linux platform application servers, the application must print using the replacement `lp` or `lpr` scripts installed by `prtinstall.en.sh`. The SGD login scripts set `PATH` to ensure that the replacement scripts take precedence over the system scripts. If the application uses a full path name, for example `/usr/bin/lp`, or modifies `PATH` itself, reconfigure the application to use `/opt/tarantella/bin/lp` or `/opt/tarantella/bin/lpr`.

Are Accounts Shared on the Application Server?

If more than one user is simultaneously logged in to the same application server with the same user name, SGD might be unable to distinguish which user owns the print jobs. SGD discards the print jobs, logging that it has done so. This occurs with UNIX or Linux system application servers that do not have an SGD printer queue.

To fix this problem, run the `prtinstall.en.sh` script to configure a printer. See [“The SGD Printer Queue Installation Script” on page 230](#).

Use the `tarantella print` command to check that print jobs from the application server printing system are reaching the SGD print queue.

Is the Windows Name of the Server the Same as the DNS Name?

If you have a Microsoft Windows NT server with a DNS name of `naples.indigo-insurance.com` and a NetBIOS name of `VESUVIUS`, print jobs from this server fail, because they contain the host identifier `VESUVIUS` instead of `naples`.

You can avoid this problem by editing the file `hostnamemap.txt` in the `/opt/tarantella/etc/data` directory. This file enables you to map host names to DNS names. The file contains instructions on how to create the mappings.

If You Are Using PDF Printing, is the Same PostScript Printer Driver Installed on Every Microsoft Windows Application Server?

To be able to use PDF printing, you must install the same PostScript printer driver on every Microsoft Windows application server.

In the Administration Console, check that the name of the driver matches the name configured in the Postscript Printer Driver field on the Global Settings → Printing tab, or the Printing tab for the user profile or parent object. The System event log on the application server shows an error if the names do not match.

SGD Server Checklist

Use the following SGD server troubleshooting steps to diagnose printing problems in SGD.

Is Printing Paused or Disabled Across the Array?

Use the `tarantella print status` command to check whether printing is paused or disabled for the array.

If necessary, enable printing, using `tarantella print start` or `tarantella print resume`.

For Printing on Microsoft Windows Client Devices, Are Client Printers Disabled?

In the Administration Console, check the Global Settings → Printing tab, or the Printing tab for the user profile or parent object. See whether users can access all their client printers, just their default client printer, or no client printers.

For PDF printing, check whether the SGD PDF printers are enabled.

Has the Array Configuration Changed?

Printing is not reconfigured if you do any of the following:

- Create an array
- Add a new secondary server to the array
- Change the primary server in the array

If the array has changed you might have to reconfigure printing, so that print jobs are sent to the correct printer.

For PDF Printing, is Ghostscript Available on the SGD Host?

PDF printing in SGD uses Ghostscript to convert print jobs into PDF files. SGD also uses Ghostscript to convert print jobs from PostScript to PCL.

If the `/opt/tarantella/var/log/print.log` file contains a message such as "Can't find ps2pdf" or "Consider obtaining Ghostscript from <http://www.ghostscript.com>", then either Ghostscript is not installed or it is installed in a non-standard location.

See [“Checking the Ghostscript Installation on the SGD Host” on page 233](#) for details of how fix Ghostscript installation problems.

Tracing a Print Job

If the checklists above do not solve your SGD printing problem, try the following troubleshooting steps. These steps enable you to track the progress of a print job from the application server to the SGD server to the client device.

Step 1: Can You Print From the SGD Server?

Configure an X or character application to run on the SGD server. Display a terminal window, for example `xterm`, and start the application from your SGD webtop.

Try printing a test page, by running the
`/opt/tarantella/bin/scripts/printtestpage.en.sh` script.

If the page does not print, run
`/opt/tarantella/bin/scripts/printtestpage.en.sh --direct`. This bypasses the UNIX or Linux system spooler.

Check the following:

■ Did the first test page print?

The problem is related to the movement of print jobs from the application server to the SGD server.

- For UNIX or Linux platform application servers, go to [“Step 3: Is the Print Job Leaving the UNIX or Linux Platform Application Server?”](#) on page 252.
- For Windows Terminal Services, go to [“Step 5: Is the Print Job Leaving the Windows Terminal Services Application Server?”](#) on page 253.

■ Did the second test page print?

The problem is related to the UNIX or Linux system printing system on the SGD host.

Investigate and fix any problems, using your UNIX or Linux system documentation for help. Then try printing again.

■ Did neither of the test pages print?

The problem is related to the SGD server.

Go to [“Step 2: Is the SGD Printer Queue Installed on the SGD Server?”](#) on page 251.

Step 2: Is the SGD Printer Queue Installed on the SGD Server?

In the list of printers on the SGD host, check for an entry for `tta_printer`.

Consult your UNIX or Linux system documentation to find out how to display the list of printers. On some systems, you can use `lpstat -t`. If your system has a file `/etc/printcap`, this contains a list of printers in plain text format.

Check the following:

- **Is the tta_printer printer present on the SGD host?**

The problem is related to the movement of print jobs from the SGD server to the client device. Go to [“Step 7: Have You Examined the Print Log Files?” on page 254.](#)

- **Is the tta_printer printer missing from the SGD host?**

Run the `prtinstall.en.sh` script on the SGD server. Then try printing again.

See also [“The SGD Printer Queue Installation Script” on page 230.](#)

Step 3: Is the Print Job Leaving the UNIX or Linux Platform Application Server?

Using an application object configured to display a terminal window on the UNIX or Linux system application server, try printing a small text file to the SGD printer. For example, type the command: `lp -d tta_printer /etc/hosts`.

Check the following:

- **Does the command return an error message?**

Check that the UNIX or Linux platform application server is configured to print through SGD. You might need to run the `prtinstall.en.sh` script. See [“The SGD Printer Queue Installation Script” on page 230](#) for more details.

- **Does the command return a print job ID?**

This suggests that SGD printing is correctly configured, but the problem might lie in the UNIX or Linux print system. Go to [“Step 4: Is the Print Job Present in the UNIX or Linux System Spool Directory?” on page 252.](#)

Step 4: Is the Print Job Present in the UNIX or Linux System Spool Directory?

The print spool directory varies between different UNIX or Linux systems. Consult your UNIX or Linux system documentation for assistance.

Check the following:

- **Is the print job present in the spool directory?**

There might be a network problem between the application server and SGD server. Go to [“Step 6: Is the Print Job Reaching the SGD Server?” on page 253.](#)

- **Is the print job missing from the spool directory?**

Check your UNIX or Linux system LPD printing configuration. For example, ensure that there are suitable entries in `/etc/hosts.equiv` or `/etc/hosts.lpd`, and that there are no `.deny` files, such as `/etc/hosts.equiv.deny`.

Check that the lpd daemon is running and listening. For example, use the following commands:

```
# ps -ef | grep lpd
# netstat -a | grep printer
```

Try printing again.

Step 5: Is the Print Job Leaving the Windows Terminal Services Application Server?

Check the print queue on the application server. Consult your system documentation if you need help on how to do this.

Check the following:

- **Is the print job leaving the application server?**

There might be a network problem between the application server and SGD server. Go to [“Step 6: Is the Print Job Reaching the SGD Server?”](#) on page 253.

- **Is the print job leaving the application server?**

Check the configuration of the SGD printer, as follows:

- Check that you can ping and telnet to the SGD server from the application server.
- Look for errors in the Event Log.
- From a command prompt, use the `lpr -s server -p tta_printer filename` command to print. If this works, the printer driver on the application server might not be installed or configured correctly.

Step 6: Is the Print Job Reaching the SGD Server?

Check the SGD print spool directories on the SGD server:

`/opt/tarantella/var/spool` and `/opt/tarantella/var/print/queue`.

Check the following:

- **Is the print job present on the SGD server?**

Check that you are using fully qualified DNS names in the application object, and that name resolution is working correctly.

Examine the printing log files for more information. Go to [“Step 7: Have You Examined the Print Log Files?”](#) on page 254.

- **Is the print job missing from the SGD server?**

Check the configuration of the SGD server, as follows:

- Check your UNIX or Linux system LPD printing configuration.

For example, ensure that there are suitable entries in `/etc/hosts.equiv` or `/etc/hosts.lpd`, and that there are no `.deny` files, such as `/etc/hosts.equiv.deny`.

Check that the `lpd` daemon is running and listening. For example, use the following commands:

```
# ps -ef | grep lpd
# netstat -a | grep printer
```

- Check that you can ping and telnet to the SGD server from the application server.
- If you are using Windows Terminal Services, display a command prompt and use the `lpr -s server -p tta_printer filename` command to print. If this works, this suggests the printer driver on the application server is not installed or configured correctly.

Step 7: Have You Examined the Print Log Files?

You can use the `tarantella query` command to examine the logs across the array. Log files are stored in `/opt/tarantella/var/log` on each SGD server in the array.

If the print log files are empty, edit the Log Filter, to log printing messages. In the Administration Console, go to the Global Settings → Monitoring tab, and add the following log filters:

```
server/printing/*:print%%PID%%.log
server/printing/*:print%%PID%%.jsl
```

If the log contains messages indicating problems with user name mappings, this suggests you are using shared accounts on the application server. See [“Are Accounts Shared on the Application Server?”](#) on page 249.

Troubleshooting Other Printing Problems

This section describes some typical problems when printing through SGD and includes the following topics.

- [“Troubleshooting Printer Preferences and Settings”](#) on page 255
- [“Print Jobs Can Be Queued When SGD Printing is Disabled”](#) on page 256
- [“Fonts Do Not Print Correctly With PDF Printing”](#) on page 257
- [“Changing Printer Names in Windows Application Sessions”](#) on page 257
- [“Changing the Names of the SGD PDF Printers”](#) on page 258

- “Users See a Printer Called ‘_Default’ in a Windows Application Session?” on page 258

Troubleshooting Printer Preferences and Settings

When printing from a Windows application, users can set preferences for the printers they use. The following are common problems with printer preferences.

- “Current Client Printer Preferences Are Ignored” on page 255
- “Changes to Printer Preferences Are Not Remembered” on page 255
- “Printer Preferences Are Corrupted” on page 255
- “Printer Preferences Are Lost When a User Changes Printers” on page 256
- “Local Printer Settings Are Not Set in the Remote Windows Application Session” on page 256
- “Printer Settings Are Ignored When Using PDF Printing” on page 256

Current Client Printer Preferences Are Ignored

The first time a client printer is defined for a user, the printer preferences, such as the paper size and orientation, are the application server’s defaults for the printer driver and not the client printer’s current preferences.

Users can change the printer preferences on the application server, and these modified preferences are used when they next connect using a client device with the same printer.

Changes to Printer Preferences Are Not Remembered

When a user changes their printer preferences, for example by changing the default paper size, sometimes the change is not remembered when they next run a Windows application.

There is a delay between changing the preferences and the new preferences being sent to the client. When changing printer preferences, it is advisable to wait a few minutes before logging out of the Windows application.

Printer Preferences Are Corrupted

If large numbers of client printers are present, printer preferences can sometimes become corrupted.

Use the Printer Preference Caching (`--noprintprefs`) attribute to disable the storing of printer preferences for a Windows application, as follows:

```
$ tarantella object edit --name appname --noprintprefs 1
```

where *appname* is the name of the Windows application.

Printer Preferences Are Lost When a User Changes Printers

Printer preferences are linked directly to the driver name. So, if a user changes the printer they use and the new printer uses a different driver name, they have to set the printer preferences again.

Local Printer Settings Are Not Set in the Remote Windows Application Session

The printer settings of a local printer are not set on the printer in the remote Windows application session when you use SGD. However, they are set when you use the Microsoft Terminal Services Client.

SGD does not support this capability.

Printer Settings Are Ignored When Using PDF Printing

If you are using PDF printing on a Microsoft Windows client device, some printer settings might be ignored by the Adobe Reader.

This might be because the printer driver used for PDF printing has settings that are not available on the client printer.

Some settings, such as page orientation, have to be set in the Adobe Reader print dialog, as well as on the printer in the Windows application session. Once you have set up the Reader, the settings are remembered.

Print Jobs Can Be Queued When SGD Printing is Disabled

After disabling the SGD print system, by running `tarantella print stop`, it is still possible to spool print jobs on application servers. These jobs remained queued until SGD printing is restarted.

To prevent print jobs from being submitted, disable the SGD print queue manually on the application servers.

Fonts Do Not Print Correctly With PDF Printing

When using PDF printing, users might find that the fonts on the printed output are not what they expected.

As PDF printing relies on a combination of Windows printer drivers, when printing from Windows applications, Ghostscript and a PDF viewer to deliver its output, you might have to experiment with the font settings for each of these components to see if this produces a better result.

TrueType Fonts and Windows Applications

When printing from a Windows application and the document contains TrueType fonts, users might find that the printer is using its own fonts, called *device fonts*, instead of the TrueType fonts. This can result in some characters being printed as “empty boxes” ([]).

The solution to this problem is to force the printer to download the TrueType fonts for printing.

Display the Print dialog in the Windows application and select Properties → Advanced. In the Graphic section, change the TrueType Font option to Download as Softfont.

Changing Printer Names in Windows Application Sessions

The names of printers created in a Windows Terminal Services session depend on the client device platform.

For Unix, Linux, and Mac OS X platform client devices, the name comes from the printer configuration file used for the client device. See [“Configuring Printing to UNIX, Linux, and Mac OS X Platform Client Devices” on page 241](#) for more details.

For Windows client devices, the name comes from the printer driver.

If you are using PDF printing, you can amend the names of the PDF printers. See [“Changing the Names of the SGD PDF Printers” on page 258](#).

Changing the Names of the SGD PDF Printers

The names of SGD PDF printers are configurable. You can amend these names as follows.

To change the PDF printer names for all users, use the following command:

```
$ tarantella config edit \  
--printing-pdfprinter name --printing-pdfviewer name
```

To change the PDF printer names for an organization, organizational unit, or user profile object, the object must also be configured to override the parent object's printing settings. Use the following command:

```
$ tarantella object edit --name object \  
--userprintingconfig true --pdfprinter name --pdfviewer name
```

Users See a Printer Called ‘_Default’ in a Windows Application Session?

Users who access Windows applications from UNIX, Linux, or Mac OS X platform client devices, might see a printer called ‘_Default’ in their Windows application session. This can be confusing to users if their client printer has a different name or they have no client printer.

This is caused by the default setting in the `printerinfo.txt` file, which is used to associate the printer driver name with a print job when printing from a Windows application.

To correct the printer name, edit the `printerinfo.txt` file.

To remove the ‘_Default’ printer name, delete the ‘_Default’ entry from the `printerinfo.txt` file.

See [“Configuring Printing to UNIX, Linux, and Mac OS X Platform Client Devices” on page 241](#), for more details about the `printerinfo.txt` file.

Client Drive Mapping

Client drive mapping (CDM) enables SGD users to access the drives on their client device from applications running on UNIX, Linux, or Microsoft Windows platform application servers.

This section describes how to configure CDM for SGD users. Common problems when using CDM in SGD are also covered, along with tips on how to fix them.

This section includes the following topics:

- [“Setting Up Client Drive Mapping” on page 259](#)
- [“Troubleshooting Client Drive Mapping” on page 267](#)
- [“Logging for CDM” on page 275](#)

Setting Up Client Drive Mapping

Setting up CDM involves the following configuration steps:

1. Configure the application servers for CDM.
 - See [“Configuring UNIX and Linux Platform Application Servers for CDM” on page 259](#).
The SGD Enhancement Module must be installed on the application server.
 - See [“Configuring Microsoft Windows Application Servers for CDM” on page 262](#).
2. Enable CDM services in SGD.
 - See [“Enabling CDM Services in SGD” on page 262](#).
3. Configure the drives you want users to access from SGD.
 - See [“Configuring the Client Drives Available to Users” on page 264](#).

Configuring UNIX and Linux Platform Application Servers for CDM

Configuring UNIX and Linux platform application servers for CDM involves the following steps:

1. Install the SGD Enhancement Module for UNIX and Linux Platforms.

The *Oracle Secure Global Desktop 4.6 Installation Guide* has details of how to install the Enhancement Module.

The *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* has details of the supported platforms for the SGD Enhancement Module.
2. Configure the Network File System (NFS) share to be used for CDM.

See [“Configuring an NFS Share for CDM” on page 260](#).

3. Start the CDM processes on the application server.

See “Starting CDM Processes on the Application Server” on page 262.

Configuring an NFS Share for CDM

Configuring an NFS share for CDM involves the following:

- Configuring a shared directory on the application server
- Configuring how client drives are displayed on UNIX platforms

Configuring a Shared Directory on the Application Server

You must have an NFS server installed and running on the application server. The NFS server must share, or export, a directory to be used for CDM. By default, the directory is `/smb`. You have to manually create and export this directory.

You can specify an alternative NFS share in the CDM configuration file, `/opt/tta_tem/etc/client.prf`. Edit the `[nfsserver/mount/mountpoint={ (/smb) }]` setting to reflect the name of the share.

The NFS share must be accessible to localhost, and users must have read and write access to it. Consult your system documentation for details of how to configure an NFS server and export a directory.

Configuring How Client Drives Are Displayed on UNIX Platforms

When CDM is enabled, the user’s client drives or file systems are available by default in the `My SGD Drives` directory in the user’s home directory. The `My SGD Drives` directory is a symbolic link to the NFS share that is used for CDM.

You can configure the name and location of the symbolic link by *adding* settings to the CDM configuration file, `/opt/tta_tem/etc/client.prf`, as follows:

- **The name of the symbolic link.** This is configured with the following setting:

```
[nfsserver/user/symlinkname={ (symlink) }]
```

The default setting is: `My SGD Drives`

For example, to change the name of the symbolic link to `Client Shares`, add the following line to the configuration file:

```
[nfsserver/user/symlinkname={ (Client Shares) }]
```

- **The directory where the symbolic link is created.** This is configured with the following setting:

```
[nfsserver/user/symlinkdir={ (dir) }]
```

The default setting is: `$HOME`

For example, to create the symbolic link in the `/tmp` directory, add the following line to the configuration file:

```
[nfsserver/user/symlinkdir={ (/tmp) }]
```

The directory can also be specified using environment variables. The variables you can use are controlled by the `nfsserver/user/envvars` setting.

For example, to create the symbolic link in the `/tmp/username` directory, add the following line to the configuration file:

```
[nfsserver/user/symlinkdir={ (/tmp/$USER) }]
```

- **Environment variables for specifying the directory where the symbolic link is created.** These are configured with the following setting:

```
[nfsserver/user/envvars={ (var) ... }]
```

The default setting is: `(USER) (HOME) (LOGNAME)`

Enclose each variable in parentheses. Do not include the dollar sign (\$) before the variable name.

The variables in the list *replace* the default variables.

For example, to be able to use the `HOME`, `USER`, `DISPLAY` and `TMPDIR` variables, add the following line to the configuration file:

```
[nfsserver/user/envvars={ (HOME) (USER) (DISPLAY) (TMPDIR) }]
```

After making any changes to the CDM configuration file, you must restart the CDM processes on the application server. See [“Starting CDM Processes on the Application Server” on page 262](#) for details of how to do this.

Starting CDM Processes on the Application Server

To start the CDM processes on the application server, log in as superuser (root) and use the following commands:

```
# /opt/tta_tem/bin/tem stopcdm
# /opt/tta_tem/bin/tem startcdm
```

Configuring Microsoft Windows Application Servers for CDM

To use a Microsoft Windows application server for CDM, drive redirection must be enabled on the application server. Drive redirection is enabled by default.

Enabling CDM Services in SGD

This section describes how to enable CDM services for an array of SGD servers.

CDM can be enabled separately for applications that run on Microsoft Windows application servers (*Windows CDM*), and for applications that run on UNIX or Linux platform application servers (*UNIX platform CDM*).

By default, Windows CDM and UNIX platform CDM are disabled.

By default, you cannot use UNIX platform CDM and run a Server Message Block (SMB) service, such as Samba, on the SGD host. See [“Running UNIX Platform CDM With Another SMB Service” on page 263](#) for details of the required configuration if you want to use UNIX platform CDM and run an SMB service on the SGD host.

When you enable CDM services, you can also enable or disable *dynamic drive mapping*. This feature provides support for “hot plugging” of removable drives, such as USB memory sticks, during a user session. Dynamic drive mapping is enabled by default for an SGD array.

▼ How to Enable SGD Client Drive Mapping Services

Changes to CDM only take effect for new user sessions. Users might have to log out of SGD and log in again to access the drives on their client device.

1. **In the Administration Console, display the Global Settings → Client Device tab.**

2. (Optional) Enable Windows CDM.

Select the Windows Client Drive Mapping check box.

3. (Optional) Enable UNIX platform CDM.

a. Select the Unix Client Drive Mapping check box.

b. Start UNIX platform CDM services on each server in the array.

Either restart all the SGD servers in the array, or use the `tarantella start cdm` command on each SGD server in the array.

If you restart the SGD servers, ensure that no users are logged in to the SGD server, and that there are no application sessions, including suspended application sessions, running on the SGD server.

4. (Optional) Enable Dynamic Drive Mapping.

Dynamic drive mapping is enabled by default for an SGD array.

Select the Dynamic Drive Mapping check box.

Running UNIX Platform CDM With Another SMB Service

In a default installation, you cannot use UNIX platform CDM and run another SMB service, such as Samba, on the SGD host. This is because they both use Transmission Control Protocol (TCP) port 139. To use CDM for UNIX and Linux platform applications, you must either disable the other SMB server or configure the host to enable more than one service to use TCP port 139.

To enable more than one service to use TCP port 139, you have to configure the SGD host to have more than one Internet Protocol (IP) address. To do this, either install another network interface card (NIC), or use IP aliasing to assign multiple IP addresses to a single NIC. This is described in [“How to Run UNIX Platform CDM and Another SMB Service on the Same Host” on page 263](#).

▼ How to Run UNIX Platform CDM and Another SMB Service on the Same Host

Repeat this procedure for each SGD server that also has an SMB service enabled.

Ensure that no users are logged in to the SGD server, and that there are no application sessions, including suspended application sessions, running on the SGD server.

1. Stop the SGD server and configure the IP addresses you want it to bind to for CDM.

Use the following command:

```
# tarantella config edit \  
--tarantella-config-cdm-externalnbtaddress ip-address ...
```

The default setting for *ip-address* is ***, which means bind to all interfaces. Separate each IP address with a space.

2. When you have configured the IP addresses, start the SGD server.
3. Configure the other SMB service, or services, to bind to a different IP address.

Configuring the Client Drives Available to Users

You configure the client drives you want users to access with the Client Drive Mapping attribute on the Client Device tab for user profiles, organizational unit, and organization objects. CDM uses inheritance. You define access to client drives at an organization level, which you can override at an organizational unit level or user profile level. By default, users have read and write access to all drives.

For Windows applications, you can configure application-specific client drive access with the Client Drive Mapping attribute on the Client Device tab for the Windows application object. This overrides any CDM settings configured for organization, organizational unit, or user profile objects. The order of precedence when configuring CDM for Windows application objects is: Windows application → user profile → organizational unit → organization.

When a user logs in to an SGD server from a *Windows client device*, information is gathered about the drives on the client device. For each available drive, the Client Drive Mapping attribute on the user profile is checked. If there is no matching client drive configured, the parent organizational unit's Client Drive Mapping attribute is checked, and so on up the organizational hierarchy to the organization object.

If a match is found, then the associated access rights are granted for that drive. The access rights for a mapped client drive are shown in brackets after the drive name: (rw) means read-write access, (ro) means read only access.

At each level in the organizational level, you configure a number of drive mapping specifications. Each of these states a client drive letter and the access rights to that drive. For example, you might specify that a user has read-write access to client drive A. The first matching entry in the list is used. Make sure the most specific settings, for example, A or B, appear before more general settings, for example, All Drives.

When a user logs in to an SGD server from a *UNIX, Linux, or Mac OS X platform client device*, the SGD Client uses a local configuration file to configure access to the client file system. See [“Configuring the Drives Available to UNIX, Linux, and Mac OS X Platform Client Devices” on page 265](#) for more details.

Note – Changes to client drive specifications only take effect for new user sessions.

Configuring the Drives Available to UNIX, Linux, and Mac OS X Platform Client Devices

By default, users with UNIX, Linux, and Mac OS X platform client devices have access to their home directory and this is mapped to a drive called *My Home*.

Users can configure which part of their client file system they can access from applications by editing the `$HOME/.tarantella/native-cdm-config` configuration file. This file is automatically created when the SGD Client is installed. The file contains detailed instructions for users on how to create mapped drives.

The [CDM] section of this configuration file contains entries of the form `<path> <type> <label>` where:

- `<path>` is the absolute path name of the client file system.
- `<type>` is either `fixed`, `floppy`, `cdrom`, `remote` or `removable`.
- `<label>` is the name that is used in the application session.

Use a separate line for each drive and separate each of the fields with a space or a tab. If either the `<path>` or the `<label>` fields contains spaces or tabs, enclose the field in quotes.

You can use environment variables in the `<path>` or `<label>` fields. You delimit these with a dollar sign (`$`). To use a literal `$`, escape it with another `$`.

The following is an example configuration file:

```
[CDM]
$HOME$ fixed "My Home"
/tmp/$USER$ fixed Temp
"/mnt/win/My Documents" fixed "My Local Documents"
...
[/CDM]
```

Note – Changes to the `native-cdm-config` configuration file only take effect for new user sessions.

The access rights for a mapped client drive are shown in brackets after the drive name: (rw) means read-write access, (ro) means read only access.

An Example of Configuring Drive Availability for Users

The following example shows how to disable access to all client drives for all users in the Indigo Insurance organization. Only a single user in the organization, Ruby Port, is allowed to access the floppy drive on her Windows computer.

In the Administration Console, go to the Client Device tab and display the Client Drive Mapping table for the o=Indigo Insurance organization object. In the Client Drive Mapping table, select the check box next to All Drives. Click the Edit button and set the Access Rights to None. This disables access to all client drives.

In the Administration Console, go to the Client Device tab and display the Client Drive Mapping table for the Ruby Port user profile object. In the Client Drive Mapping table, click the New button and configure the following settings:

- **Client Device Drive.** Select A:, the drive letter of Ruby's floppy drive, or R/W Removable. R/W Removable matches all read-write removable drives, such as floppy drives.
- **Access Rights.** Select Read/Write. This gives Ruby full access to the drive, as long as the floppy disk is not write-protected.

This gives Ruby Port full access to the floppy drive on her Windows computer.

Detecting Removable Drives

If users attach a removable drive such as a Universal Serial Bus (USB) memory stick during a user session, SGD detects and mounts the device automatically. This feature is called dynamic drive mapping.

Dynamic drive mapping is enabled by default for an SGD array.

To use dynamic drive mapping, either Windows CDM or UNIX platform CDM must be enabled.

On UNIX and Linux platform client devices that do not use hardware abstraction layer (HAL) to detect client drives, SGD looks for removable drives by monitoring the locations listed in the [DYNAMICSTORAGE] section of the

`$HOME/.tarantella/native-cdm-config` configuration file. Depending on the client platform, the following default system locations are listed in the `[DYNAMICSTORAGE]` section of this file.

| Client Platform | Location | Type |
|-----------------|--------------------------------|-----------|
| Linux | /media | removable |
| Solaris OS | /rmdsk | removable |
| | /cdrom | cdrom |
| Sun Ray | <code>\${DTDEVROOT}/mnt</code> | removable |
| Mac OS X | /Volumes | removable |

You can specify additional directories to be monitored, by adding one or more entries to the `[DYNAMICSTORAGE]` section. For example, the following entry causes SGD to monitor the `/opt` directory for removable drives, in addition to the default location for the client platform.

```
[DYNAMICSTORAGE]
...
/opt removable
[/DYNAMICSTORAGE]
```

On Mac OS X client devices, virtual drives (`.dmg` files) are not detected as dynamic drives.

Troubleshooting Client Drive Mapping

The following are common problems when using CDM in SGD:

- [“For UNIX Platform CDM, No Client Drives Are Mapped Within the User’s Session or There Are Fewer Drives Than Expected” on page 268](#)
- [“For Windows CDM, No Client Drives Are Mapped Within the User’s Session or There Are Fewer Drives Than Expected” on page 271](#)
- [“Removable Drives Attached During a User Session are Not Detected Automatically” on page 272](#)
- [“Invalid Password Errors on Microsoft Windows Application Servers” on page 273](#)
- [“More Client Drives Are Mapped Than Expected” on page 274](#)
- [“The Recycle Bin Does Not Work As Expected” on page 274](#)
- [“Mapped Drives Have Unusual Names” on page 274](#)
- [“CDM Limitations for Shared Users” on page 274](#)

- [“Disabling CDM for a Client Device” on page 275](#)
- [“Logging for CDM” on page 275](#)

For UNIX Platform CDM, No Client Drives Are Mapped Within the User’s Session or There Are Fewer Drives Than Expected

Use the following checklist to resolve this problem.

Is the SGD Enhancement Module installed on the application server?

To access client drives from UNIX or Linux platform applications displayed through SGD, the SGD Enhancement Module must be installed on the application server.

The *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* has details of the supported platforms for the SGD Enhancement Module.

Is UNIX platform CDM enabled?

In the Administration Console, go to the Global Settings → Client Device tab and ensure that the Unix Client Drive Mapping check box is selected.

Remember, UNIX platform CDM services only become available when you restart all SGD servers in the array. To manually start CDM services without restarting the array, run the `tarantella start cdm` command on all members of the array.

Have the user’s client drives been configured correctly?

The Client Drive Mapping attribute on the Client Device tab for organization, organizational unit, and user profile objects determines which client drives each user can access. The user might be configured to have no access to any client drives. Remember to check the ancestor OUs in the organizational hierarchy. CDM settings are inherited, so you can give access to many users with one configuration change.

For Windows applications, application-specific client drive access can be configured using the Client Drive Mapping attribute on the Client Device tab for the Windows application object. Remember that this overrides any CDM settings configured for organization, organizational unit, or user profile objects.

For users with *UNIX, Linux, or Mac OS X* platform client devices, check that the user’s `$HOME/.tarantella/native-cdm-config` file is present and has valid entries. If this file is not present, a default version is created automatically when the user next logs in to SGD.

Are UNIX platform CDM processes running?

On the host where SGD is installed, use the following command:

```
# ps -ef | grep ttacdmd
```

If UNIX platform CDM processes are running, there are at least two processes with the name `ttacdmd`.

If there are no any drive mapping processes, use the following command:

```
# grep cdm /opt/tarantella/var/log/*
```

Check the output for any messages.

On UNIX and Linux platform application servers, use the following command to check that CDM processes are running:

```
# /opt/tta_tem/bin/tem status
```

If CDM processes are not running, use the following command:

```
# /opt/tta_tem/bin/tem startcdm
```

If starting CDM processes produces errors such as "Failed to mount /smb", check that the NFS server is running and that the directory being used for CDM is exported correctly.

Check whether another service is using port 4242. If so, edit the `/opt/tta_tem/etc/client.prf` file and change the port number in the line `[nfsserver/mount/port={ (4242) }]` and restart the CDM processes.

Are you using a proxy server?

Proxy servers drop a connection after a short period of time if there is no activity on the connection.

SGD sends keepalive packets to keep the connection open between the client device and the SGD server and by default this is every 100 seconds. This connection is used for CDM. Try increasing the frequency of the keepalive packets.

See also [“Proxy Server Timeouts” on page 11](#).

Do the version numbers for the SGD Enhancement Module and the SGD server match?

Run the following command on the host where SGD is installed:

```
$ tarantella version
```

Make a note of the version number.

On UNIX and Linux platform application servers, run the following command:

```
$ /opt/tta_tem/bin/tem version
```

Are other services using TCP port 139?

UNIX platform CDM services must bind to TCP port 139, which is used for SMB services. This port might already be in use, for example by a product such as Samba.

To find out whether any other process is using port 139, stop the SGD server and then run the following commands on the host where SGD is installed:

```
$ netstat -an | grep 139
$ grep 139 /etc/xinetd.conf
```

To ensure that UNIX platform CDM services are available, stop any other products that bind to TCP port 139 and restart the SGD server.

Follow the instructions in [“How to Run UNIX Platform CDM and Another SMB Service on the Same Host”](#) on page 263.

Have all the client drives been found?

For Windows client devices, the SGD Client displays information about the drives it has found. Click the right mouse button on the System Tray icon and select Connection Info.

For UNIX and Linux platform client devices, this information is written to the SGD Client log file.

For users with UNIX or Linux platform client devices, check that the user’s `$HOME/.tarantella/native-cdm-config` file is present and has valid entries. If a client drive is not being detected automatically, add an entry to the [CDM] section of this file to specify the location where the drive is mounted.

Does logging reveal any errors?

On UNIX or Linux platform application servers, check for any drive mapping errors in the `clerr.log` and the `clPID.log` files in the `/opt/tta_tem/var/log` directory.

See also [“Logging for CDM”](#) on page 275.

Is the drive mapping connection between the UNIX or Linux platform application server and the SGD server working?

On UNIX or Linux platform application servers, drive mapping errors are reported to the `clerr.log` and the `clPID.log` files in the `/opt/tta_tem/var/log` directory. See also [“CDM Diagnostics for UNIX or Linux Platform Application Servers” on page 275](#).

For Windows CDM, No Client Drives Are Mapped Within the User’s Session or There Are Fewer Drives Than Expected

Use the following checklist to resolve this problem.

Is drive redirection enabled on the Microsoft Windows application server?

Drive redirection is enabled by default on Microsoft Windows application servers.

To access client drives from Windows applications displayed through SGD, the SGD Enhancement Module does not have to be installed on the application server.

Is Windows CDM enabled?

In the Administration Console, go to the Global Settings → Client Device tab and ensure that the Windows Client Drive Mapping check box is selected.

Have the user’s client drives been configured correctly?

The Client Drive Mapping attribute on the Client Device tab for organization, organizational unit, and user profile objects determines which client drives each user can access. The user might be configured to have no access to any client drives. Remember to check the ancestor OUs in the organizational hierarchy. CDM settings are inherited, so you can give access to many users with one configuration change.

For Windows applications, application-specific client drive access can be configured using the Client Drive Mapping attribute on the Client Device tab for the Windows application object. Remember that this overrides any CDM settings configured for organization, organizational unit, or user profile objects.

For users with *UNIX*, *Linux*, or *Mac OS X* platform client devices, check that the user’s `$HOME/.tarantella/native-cdm-config` file is present and has valid entries. If this file is not present, a default version is created automatically when the user next logs in to SGD.

Are Windows CDM processes running?

On Microsoft Windows application servers, use the Terminal Services Manager to check that there is an RDP session for the user.

Are you using a proxy server?

Proxy servers drop a connection after a short period of time if there is no activity on the connection.

SGD sends keepalive packets to keep the connection open between the client device and the SGD server and by default this is every 100 seconds. This connection is used for CDM. Try increasing the frequency of the keepalive packets.

See also [“Proxy Server Timeouts” on page 11](#).

Have all the client drives been found?

For Windows client devices, the SGD Client displays information about the drives it has found. Click the right mouse button on the System Tray icon and select Connection Info.

For UNIX and Linux platform client devices, this information is written to the SGD Client log file.

For users with UNIX or Linux platform client devices, check that the user’s `$HOME/.tarantella/native-cdm-config` file is present and has valid entries. If a client drive is not being detected automatically, add an entry to the [CDM] section of this file to specify the location where the drive is mounted.

Does logging reveal any errors?

On Microsoft Windows application servers, check the Windows Event Viewer for any drive mapping errors.

See also [“Logging for CDM” on page 275](#).

Is the drive mapping connection between the Microsoft Windows application server and the SGD server working?

To check whether the drive mapping connection between the application server and the SGD server is working, use the Terminal Services Manager on the Microsoft Windows application server to check that there is an RDP session for the user. See also [“CDM Diagnostics for Microsoft Windows Application Servers” on page 275](#).

Removable Drives Attached During a User Session are Not Detected Automatically

Check that dynamic drive mapping is enabled for the SGD array.

```
$ tarantella config list --array-dyndevice
```


By default, dynamic drive mapping is enabled.

To use dynamic drive mapping, CDM must be enabled for the array. Go to the Global Settings → Client Device tab in the Administration Console and check that Windows Client Drive Mapping or Unix Client Drive Mapping is enabled.

The Client Drive Mapping attribute on the Client Device tab for organization, organizational unit, and user profile objects determines which client drives each user can access. The user might be configured to have no access to any client drives. Remember to check the ancestor OUs in the organizational hierarchy. CDM settings are inherited, so you can give access to many users with one configuration change.

For Windows applications, application-specific client drive access can be configured using the Client Drive Mapping attribute on the Client Device tab for the Windows application object. Remember that this overrides any CDM settings configured for organization, organizational unit, or user profile objects.

For users with *UNIX, Linux, or Mac OS X* platform client devices, check that the user's `$HOME/.tarantella/native-cdm-config` file is present and has valid entries. If a removable drive is not being detected automatically, add an entry to the `[DYNAMICSTORAGE]` section of this file to specify the location where the drive is mounted.

Check the SGD Client log file for information about whether attached drives have been detected correctly. Set the SGD Client logging level to All to log verbose messages about dynamic drive mapping.

Invalid Password Errors on Microsoft Windows Application Servers

If no client drives are mapped in the Microsoft Windows application session and you see errors such as `Add device failed with ERROR_INVALID_PASSWORD` in the CDM log output, this can be caused by the LAN Manager authentication level setting. The LAN Manager authentication level controls the authentication protocols used for communications between a client and Microsoft Windows server. If the authentication level is set too high, CDM fails.

The solution is to edit the `Security options\Network security\LAN Manager authentication level` policy and select `Send LM & NTLM - Use NTLMv2 session security if negotiated`.

See [Microsoft KB article 823659](#) for more details.

See also “[Logging for CDM](#)” on page 275.

More Client Drives Are Mapped Than Expected

Client drives are inherited within the organizational hierarchy, so you can give access to many users with one configuration change. Check the Client Drive Mapping attribute on the organizational unit object that the user profile object belongs to. If necessary, check all ancestors of the user profile, including the top-level organization object. You can override a setting that is specified in a parent organizational unit (OU) or organization object, by configuring the user profile's Client Drive Mapping attribute. The first matching drive specification is used.

For users with *UNIX, Linux, or Mac OS X* platform client devices, check that the user's `$HOME/.tarantella/native-cdm-config` file is present and has valid entries.

The Recycle Bin Does Not Work As Expected

On Microsoft Windows client devices, client drives accessed through SGD are treated by the application server as network drives. This means that Recycle Bin features are not available for client drives.

Deleting a file does not send the file to the Recycle Bin. The `Recycled` directory, if present, is not shown as the Recycle Bin, and its contents are not displayed.

Mapped Drives Have Unusual Names

On Microsoft Windows client devices, the names of mapped drives are of the form *clientdrive (access) on clientname*. For example:

| |
|----------------------|
| C (rw) on MYCOMPUTER |
|----------------------|

On UNIX, Linux, and Mac OS X platform client devices, the names of mapped drives are configured in the user's `$HOME/.tarantella/native-cdm-config` file. Check that this file has valid entries.

CDM Limitations for Shared Users

On Unix or Linux platform application servers, access to client file systems is given to users based on their UNIX system user ID and standard NFS file system privileges. If a shared account is used to access applications, CDM is not available. This is because SGD has no way to distinguish between these users, as they all have the same user ID.

Disabling CDM for a Client Device

For security purposes, you may want to disable CDM for a client device.

To disable CDM for the client device, edit the `<enablecdm>` entry in the `<localsettings>` section of the client profile. A setting of 0 disables CDM.

Logging for CDM

Logging can be used to diagnose problems with CDM. You configure and use logging for the SGD array and for application servers, as follows:

- Enable CDM logging for the SGD array
- Use CDM diagnostics for Microsoft Windows application servers
- Use CDM diagnostics for UNIX or Linux platform application servers
- Use SGD Client logging for the client device

Enabling CDM Logging for the SGD Array

Add the following filters in the Log Filters field on the Monitoring tab of the Administration Console.

```
cdm/*/*:cdm%%PID%.jsl  
cdm/*/*:cdm%%PID%.log  
server/deviceservice/*:cdm%%PID%.log  
server/deviceservice/*:cdm%%PID%.jsl
```

CDM Diagnostics for Microsoft Windows Application Servers

On Microsoft Windows application servers, drive mapping errors are written to the Windows Event Viewer.

CDM Diagnostics for UNIX or Linux Platform Application Servers

On UNIX or Linux platform application servers, drive mapping errors are reported to the `clerr.log` and the `clPID.log` files in the `/opt/tta_tem/var/log` directory.

SGD Client Logging for Client Devices

By default, the SGD Client logs any warning messages about client drives. Log messages are stored to a file, `ccc.txt` on the client device.

To record detailed information about CDM, set the SGD Client logging level to All. See [“SGD Client Logging” on page 407](#) for more information about configuring and using SGD Client logging.

For Windows client devices, the SGD Client displays information about the drives it has found. Click the right mouse button on the System Tray icon and select Connection Info. CDM information is written to the SGD Client log file.

For UNIX and Linux platform client devices, CDM information is written to the SGD Client log file.

Audio

This section describes how to configure SGD audio services for Windows applications and X applications. Troubleshooting information for SGD audio is also included.

The following topics are covered:

- [“Setting Up Audio” on page 276](#)
- [“Configuring Microsoft Windows Application Servers for Audio” on page 277](#)
- [“Configuring UNIX and Linux Platform Application Servers for Audio” on page 277](#)
- [“Enabling SGD Audio Services” on page 279](#)
- [“Configuring Client Devices for Audio” on page 280](#)
- [“Troubleshooting Audio in Applications” on page 281](#)

Setting Up Audio

Setting up audio involves the following configuration steps:

1. Configure the application servers for audio.
 - Configure Microsoft Windows application servers.

Audio redirection must be configured on the Microsoft Windows application server.

See [“Configuring Microsoft Windows Application Servers for Audio”](#) on page 277.

- Configure UNIX and Linux platform application servers.

Configure the audio module of the SGD Enhancement Module on the UNIX or Linux platform application server.

See [“Configuring UNIX and Linux Platform Application Servers for Audio”](#) on page 277

2. Configure X application objects to use the correct audio device and audio format.

See [“Configuring X Applications for Audio”](#) on page 279.

3. Enable the SGD audio services.

See [“Enabling SGD Audio Services”](#) on page 279.

4. Configure the client device to play audio.

See [“Configuring Client Devices for Audio”](#) on page 280.

Configuring Microsoft Windows Application Servers for Audio

You can only play audio if audio redirection is enabled on the Windows Terminal Server. See [“Configuring Microsoft Windows Terminal Services for Use With SGD”](#) on page 140 for details of the Windows platforms that support audio redirection.

Configuring UNIX and Linux Platform Application Servers for Audio

To be able to hear audio in an X application, you must install and run the audio module of the SGD Enhancement Module on the UNIX or Linux platform application server.

Installing the Audio Module

See the *Oracle Secure Global Desktop 4.6 Installation Guide* for instructions on installing the audio module. If you did not install the audio module when you installed the SGD Enhancement Module, you must uninstall and then reinstall the SGD Enhancement Module.

Note – If you are using zones on Solaris™ Operating System (Solaris OS) platforms, the audio module must be installed in the global zone.

The audio module installs the SGD audio daemon and audio driver emulator. On Linux platforms, the audio driver emulator requires the `soundcore` module in the kernel. The audio driver emulator is an Open Sound System (OSS) emulator.

Note – As the audio module includes an audio driver emulator, the application server itself does not actually need to have a sound card.

Starting the Audio Module

If the audio module is installed, you start the audio service with the `/opt/tta_tem/bin/tem_startaudio` command. You must be superuser (root) to use this command.

About the SGD Audio Daemon

When audio is enabled and the user starts an X application, the SGD login script starts the SGD audio daemon, `sgdaudio`, on the application server.

The audio daemon connects to an SGD audio driver emulator, `sgdadem`, and starts an audio device node in the `/tmp/SGD/dev/sgdaudio` directory. The audio daemon sets the `SGDAUDIODEV`, `AUDIODEV`, and `AUDIO` environment variables to the location of the audio device node. The audio device node is then used to play audio during the application session.

The audio daemon transfers the audio data to the SGD server, which then sends the data to the client.

The audio daemon supports the following audio data formats:

- u-law and A-law with 8-bit precision
- 16-bit linear Pulse-code modulation (PCM)

To play audio, the client device must also support these formats.

The audio daemon supports any sample rate from 8000 Hz to 48 kHz for one or two channels. The audio daemon uses the sample rate specified by the UNIX Audio Sound Quality attribute on the Global Settings → Client Device tab in the Administration Console. By default, the sample rate is 22.05kHz.

The SGD audio daemon connects to the SGD server on random ports. If there is a firewall between the application server and the SGD server, the firewall must allow connections on all ports from the application server to the SGD server.

Configuring X Applications for Audio

To be able to hear audio in an X application, the X application might have to be configured to output audio using the right audio device and audio format.

Some X applications are hard-coded to use the `/dev/audio` or `/dev/dsp` devices for audio output. You can enable an SGD audio redirection library, to force the X application to use the device specified by the `SGDAUDIODEV` environment variable.

In the Administration Console, go to the Client Device tab for the X application and select the Audio Redirection Library check box.

Alternatively, use the following command:

```
$ tarantella object edit --name obj --unixaudiopreload true
```

As the SGD audio driver emulator is an OSS driver, the X application might have to be configured to use OSS. If your system uses the Advanced Linux Sound Architecture (ALSA), you might have to enable the ALSA OSS emulation modules in the kernel.

If the Connection Method (`--method`) used for the X application is SSH and the application's Window Type (`--displayusing`) is Kiosk, the Session Termination (`--endswhen`) attribute must be set to Login Script Exit or No Visible Windows (`--loginscriptnowindows`).

Enabling SGD Audio Services

To be able to hear audio in Windows applications and X applications, audio services must be enabled for the SGD array.

Firewalls between SGD servers can interfere with the connections required for Windows audio, see [“Firewalls Between SGD Servers” on page 14](#).

▼ How to Enable the SGD Windows Audio Service

To be able to hear audio in a Windows application, the SGD Windows audio service must be enabled in the array. The Windows audio service is disabled by default.

1. In the Administration Console, go to the Global Settings → Client Device tab and select the Windows Audio check box.

Tip – You can also use the `tarantella config edit --array-audio` command to enable the SGD Windows audio service.

The audio service only takes effect for new user sessions. Users must log out of SGD and log back in again to enable audio in their current Windows Terminal Server sessions.

2. (Optional) Set the audio quality.

Select an option for Windows Audio Sound Quality.

The default is Medium Quality Audio, using a sample rate of 22.05kHz. Only change this setting if you experience problems with audio quality.

▼ How to Enable the SGD UNIX Audio Service

To be able to hear audio in an X application, the SGD UNIX audio service must be enabled in the array. The UNIX audio service is disabled by default.

1. In the Administration Console, go to the Global Settings → Client Device tab and select the Unix Audio check box.

Tip – You can also use the `tarantella config edit --array-unixaudio` command to enable the SGD UNIX audio service.

Note – The audio service only takes effect for new user sessions. Users must log out of SGD and log back in again to enable audio in their X application sessions.

2. (Optional) Set the audio quality.

Select an option for Unix Audio Sound Quality.

The default is Medium Quality Audio, using a sample rate of 22.05kHz. Only change this setting if you experience problems with audio quality.

Configuring Client Devices for Audio

To be able to hear audio in an Windows application or X application, the client device must be capable of playing audio.

Users with Solaris OS or Linux platform client devices must have read and write access to the following audio devices:

- The `/dev/audio` device on Solaris OS platforms
- The `/dev/dsp` device on Linux platforms

For Linux platform client devices, the Enlightened Sound Daemon, also known as ESD or Esound, must be running on the client device.

ESD is usually started when the client device desktop session is started. Otherwise, the daemon must be autospawned by the ESD library on request. Ensure that autospawning is enabled in the ESD configuration file, `/etc/esd.conf`. The correct setting is `auto_spawn=1`.

Audio mixing on the client device is supported. On Solaris OS workstations, Microsoft Windows, and Mac OS X client devices, the client hardware performs the mixing. On Linux and SunRay client devices, ESD is required to perform mixing.

Troubleshooting Audio in Applications

The following are common problems when using audio in Windows applications and X applications:

- [“No Audio Plays At All” on page 281](#)
- [“Audio Is Muffled or Distorted” on page 286](#)
- [“Not All Users Require Audio” on page 286](#)
- [“Enabling UNIX Audio Debug Logging” on page 286](#)

No Audio Plays At All

If no audio is playing at all in the application session, use the following checklist to resolve the problem.

For *Windows applications and X applications*, you can use the following checklist.

Does the client device have an audio device?

To be able to play audio, the client device must have an audio device. If there is an audio device, check that the audio device works.

Users with Solaris OS or Linux platform client devices must also have read and write access to the following audio devices:

- The `/dev/audio` device on Solaris OS platforms
- The `/dev/dsp` device on Linux platforms

Note – On Solaris OS platforms, if the `AUDIODEV` environment variable has been set to a different device, the SGD Client attempts to use this device before using the `/dev/audio` device.

For Linux platform client devices, is ESD running?

For Linux platform client devices, ESD must be running.

Use the following command to check if ESD is running:

```
$ ps -ef | grep esd
```

ESD is usually started when the client device desktop session is started. If ESD is not running, check that autospawning is enabled in the ESD configuration file, `/etc/esd.conf`. The correct setting is `auto_spawn=1`.

Is the volume muted on the client device?

Check the volume control on the client device, to see whether the user has muted the volume or set the volume level too low to hear.

Is the volume muted on the application server?

Check the volume control on the application server, or in the application, to see whether the user has muted the volume or set the volume level too low to hear.

Has the audio service been enabled on the SGD server?

By default, SGD audio services are disabled for an SGD array.

See [“How to Enable the SGD Windows Audio Service” on page 279](#) for details of how to enable the SGD Windows audio service.

See [“How to Enable the SGD UNIX Audio Service” on page 280](#) for details of how to enable the SGD UNIX audio service.

Has the audio quality been changed?

By default, the SGD audio service uses Medium Quality Audio. Changing the audio quality to Low Quality Audio or High Quality Audio limits the audio formats used in the application session and might mean that the client device cannot play audio.

Reset the audio quality to Medium Quality Audio on the Global Settings → Client Device tab in the Administration Console.

For Windows applications, is audio redirection enabled on the application server?

You can only play audio if audio redirection is enabled on the Windows Terminal Server. See [“Configuring Microsoft Windows Terminal Services for Use With SGD” on page 140](#) for details of the Windows platforms that support audio redirection.

Audio redirection is disabled by default on Windows Terminal Servers.

For Windows applications, is the Remote Audio attribute enabled?

The Remote Audio (`--remoteaudio`) attribute for a Windows application object causes audio to be played on the Windows application server, rather than the client device. This attribute is disabled by default for a Windows application object.

In the Administration Console, you disable the Remote Audio attribute on the Client Device tab for the Windows application object.

For Windows applications, is there a firewall between the SGD server hosting the user session and the SGD server hosting the application session?

For Windows applications, firewalls between SGD servers can interfere with audio connections, see [“Firewalls Between SGD Servers” on page 14](#).

For X applications, is there a firewall between the application server and the SGD server?

For X applications, the SGD audio daemon connects to the SGD server on random ports. If there is a firewall between the application server and the SGD server, the firewall must allow connections on all ports from the application server to the SGD server.

For X applications, are you running compatible versions of SGD and the SGD Enhancement Module?

UNIX audio services might not work correctly if the versions of SGD and the SGD Enhancement Module are different. For example, to play audio in X applications displayed through SGD version 4.6, Linux and UNIX platform application servers must be running version 4.6 of the Enhancement Module.

Use the following command to check the current version of the SGD Enhancement Module:

```
$ /opt/tta_tem/bin/tem version
```

Use the following command to check the current version of SGD:

```
$ tarantella version
```

See the *Oracle Secure Global Desktop 4.6 Installation Guide* for details of how to upgrade the SGD Enhancement Module.

For X applications, have you installed the audio module of the SGD Enhancement Module?

To be able to play sound in X applications, you must install and run the audio module of the SGD Enhancement Module on the application server.

See the *Oracle Secure Global Desktop 4.6 Installation Guide* for details of how to install the SGD Enhancement Module.

Note – If you are using zones on Solaris OS platforms, the audio module only works if it is installed in the global zone.

Use the following command to check that UNIX audio processes are running:

```
$ /opt/tta_tem/bin/tem status
```

You start the UNIX audio module with the following command:

```
# /opt/tta_tem/bin/tem startaudio
```

You must be superuser (root) to use this command.

Is the X application hard-coded to use either the /dev/audio or the /dev/dsp device?

If an application is hard-coded to use either the /dev/audio or the /dev/dsp device, you might have to enable the SGD audio redirection library to ensure that the SGD audio driver emulator is used by the application. See [“Configuring X Applications for Audio” on page 279](#).

Is the X application outputting sound in the right format?

The SGD audio driver emulator is an OSS driver. The X application might have to be configured to use OSS. If your system uses ALSA, you might have to enable the ALSA OSS emulation modules in the kernel.

For UNIX or Linux platform application servers, is the SGD audio driver loaded in the kernel?

When you install the SGD Enhancement Module on the application server, you install the SGD audio driver, `sgdadem`. Check that the audio driver is loaded in the kernel.

- On Solaris OS platforms, use the `modinfo -c` command to check whether the `sgdadem` module is loaded.
- On Linux platforms, use the `lsmod` command to check whether the `sgdadem` and `soundcore` modules are loaded.

If the audio driver is installed but not loaded, you can try to load the module manually, as follows:

- On Solaris OS platforms, use the `modload -i moduleID` command. Use the `modinfo -c` command to find the *moduleID*.
- On Linux platforms, use the `modprobe sgdadem` command.

If loading the audio driver manually produces any errors, try to correct those errors and load the driver again.

If the SGD audio driver is not listed, check the audio module installation log for any errors. The installation log is

`/opt/tta_tem/var/log/tem_unixaudio_inst.log`. If the log reports any errors, try to correct those errors and load the driver again.

If the audio driver does not load into the kernel, contact Oracle Support.

For X applications, is the SGD audio daemon running on the application server?

There is an SGD audio daemon, called `sgdaudio`, running for each X application accessed through SGD. Use the following command to see the instances of the audio daemon:

```
$ ps -ef | grep -i sgdaudio
```

If the user does not have an audio daemon, check the audio daemon log files for any errors. The SGD audio daemon logs all fatal errors to the `/opt/tta_tem/var/log/sgdaudioPID.log` file.

For X applications, is there an SGD audio device node?

If the SGD audio daemon is running, it starts an audio device node in the `/tmp/SGD/dev/sgdaudio` directory.

In the X application session, check the value of the user's `SGDAUDIODEV`, `AUDIODEV` and `AUDIO` environment variables. These must be set to the location of the SGD audio device node.

If the environment variables are set correctly, check that the device file is present in the `/tmp/SGD/dev/sgdaudio` directory.

For X applications, does audio debug logging show any errors with the application?

Enable UNIX audio debug logging on the application server and check the log files for errors.

See [“Enabling UNIX Audio Debug Logging” on page 286](#) for more details.

Audio Is Muffled or Distorted

If audio is muffled or distorted, adjust the audio quality and audio compression settings to see if this improves the audio. You can adjust the following:

- The Sound Quality attribute on the Global Settings → Client Device tab in the Administration Console
- The Packet Compression attribute on the Protocol Engines → Audio tab for an SGD server in the Administration Console

Note – The net gain of compressing audio data, which is precompressed, is limited.

Not All Users Require Audio

If you enable audio on the Windows application server and enable the SGD audio service, all users can play audio in their Windows Terminal Services session. However, playing audio increases the amount of network bandwidth used and so you might want to restrict its use. Currently, the only way to do this is to disable audio for groups of users on the Windows application server. To do this you have to disable the Allow audio redirection setting for the group policy object, at Computer Configuration\Administrative Templates\Windows Components\Terminal Services\Client Server Redirection.

Changes to this setting only apply to new Windows Terminal Server sessions.

Enabling UNIX Audio Debug Logging

To enable UNIX audio debug logging, log in as superuser (root) on the application server and edit the `/etc/sgdtem.conf` file. Change the value of the `SGDUNIXAUDIODEBUG` environment variable in this file, as follows:

```
SGDUNIXAUDIODEBUG=1; export SGDUNIXAUDIODEBUG
```

To obtain debug logging output, the user must start a new instance of the application. Suspending and resuming the application does not generate any output, as this does not start a new instance of the SGD audio daemon.

The debug logging output goes to the
`/opt/tta_tem/var/log/sgdaudioPID.log` file.

Copy and Paste

This section describes how to configure and control access to copy and paste for applications displayed through SGD. Common problems with copy and paste are also described.

This section includes the following topics:

- [“Using Copy and Paste” on page 287](#)
- [“Controlling Copy and Paste in Applications” on page 288](#)
- [“An Example of Using Clipboard Security Levels” on page 289](#)
- [“Tips on Configuring Copy and Paste” on page 290](#)
- [“Copy and Paste Troubleshooting” on page 291](#)

Using Copy and Paste

Users can copy and paste *text* between applications displayed through SGD. Users can also copy and paste text between applications running on a client device and applications displayed through SGD. SGD supports the copy and paste of Unicode characters.

Users can only copy and paste *graphics* to or from Microsoft Windows 2000 or later applications.

For *Windows applications* and *X applications*, you copy and paste by using the normal method for the application you are copying from, and then the normal method for the application you are pasting to.

For *character applications*, click with the right mouse button, and then choose Copy or Paste as appropriate. To select a column of text in a character application, hold down the Shift key while selecting the text.

If a user attempts a copy and paste operation that is not permitted, for example because of differing security levels, they paste the following message instead of the copied data: Oracle Secure Global Desktop Software: Copied data not available to this application

SGD Administrators have full control over copy and paste operations in Windows applications and X applications. See [“Controlling Copy and Paste in Applications” on page 288](#).

Controlling Copy and Paste in Applications

In the Administration Console, you can control copy and paste operations for Windows applications and X applications displayed through SGD by doing the following:

- Configuring global copy and paste settings for the SGD array
- Configuring copy and paste for specific users
- Configuring copy and paste for specific applications

Configuring Global Copy and Paste Settings for the SGD Array

On the Global Settings → Client Device tab, copy and paste for SGD as a whole can be enabled or disabled. By default, copy and paste is enabled.

The Client’s Clipboard Security Level attribute can be used to assign a security level to the SGD Client. Data can only be copied from SGD to applications running on the client device if the SGD Client has the same security level or higher as the source application. This enables SGD Administrators to secure the flow of data outside of SGD. The default Client’s Clipboard Security Level is 3.

Configuring Copy and Paste for Specific Users

On the Client Device tab for organization, organizational unit, or user profile objects, the Copy and Paste attribute can be used to control which users in the organization are allowed to use copy and paste.

The setting for this attribute can be inherited from a parent object in the organizational hierarchy, so that SGD Administrators can enable or disable copy and paste for many users without having to edit each user profile object. By default, copy and paste is enabled.

Configuring Copy and Paste for Specific Applications

On the Client Device tab for Windows application and X application objects, the Copy and Paste attribute can be used to enable or disable copy and paste operations to or from the application.

The application can also be assigned a Clipboard Security Level. Users can only copy and paste data to an application displayed through SGD if the application has the same security level or higher as the source application. The source application is the application that the data was copied from. This enables SGD Administrators to secure the data available through particular applications. The default security level is 3.

When configuring security levels, the higher the number, the higher the security level.

Note – Character applications displayed through SGD are treated the same as applications running on the client. This is because character applications use the local client clipboard for copy and paste operations.

An Example of Using Clipboard Security Levels

In this example, copy and paste has been enabled for all users in an organization. The Client’s Clipboard Security Level attribute is set to 3, the default setting. The following table shows the security levels for applications displayed through SGD.

| Application | Application’s Clipboard Security Level |
|-------------|--|
| XFinance | 3 |
| XClaim | 4 |
| Write-o-Win | 4 |
| Slide-o-Win | 2 |

When an SGD user runs these applications, the following copy and paste operations are allowed.

| In This Application | An SGD User Can Paste Data From These Applications |
|---------------------|---|
| XFinance | <ul style="list-style-type: none">• Slide-o-Win. It has a lower security level.• Applications running on the client device. The client device has equal security level. |
| XClaim | <ul style="list-style-type: none">• XFinance and Slide-o-Win. They have a lower security level.• Applications running on the client device. The client device has a lower security level.• Write-o-Win. It has an equal security level. |
| Write-o-Win | <ul style="list-style-type: none">• XFinance and Slide-o-Win. They have a lower security level.• Applications running on the client device. The client device has a lower security level.• XClaim. It has an equal security level. |
| Slide-o-Win | <ul style="list-style-type: none">• Copy and paste is not allowed. All applications and the client device have a higher security level. |

Tips on Configuring Copy and Paste

The following are some tips for SGD Administrators who need to configure copy and paste settings for SGD objects.

- To disable copy and paste from applications running on the client device to all applications displayed through SGD, the value of the Client's Clipboard Security Level attribute must be *higher than* the highest value of the Application's Clipboard Security Level attribute of any application in the organizational hierarchy.
- To disable copy and paste from all applications displayed through SGD to applications running on the client device, the value of the Client's Clipboard Security Level attribute must be *lower than* the lowest value of the Application's Clipboard Security Level attribute of any application in the organizational hierarchy.
- To disable all copy and paste operations to or from the client device, deselect the Copy and Paste check box on the Global Settings → Client Device tab in the Administration Console.
- To disable all copy and paste operations for an individual Windows application or X application accessed through SGD, deselect the Copy and Paste check box on the Client Device tab for the application in the Administration Console.

- Inherit the copy and paste settings from other objects in the organizational hierarchy as much as possible. Only enable or disable copy and paste for individual users if you really have to. This simplifies the administration of copy and paste settings.
- For best results when copying and pasting non-ASCII text, run SGD in a UTF-8 locale. If it is not possible to do this and UTF-8 locales are installed on the SGD host, you can specify a UTF-8 locale by setting the `TTA_TEXTCONV_LANG` environment variable. For example:

```
TTA_TEXTCONVLANG=en_US.UTF8; export TTA_TEXTCONVLANG
```

You must restart SGD for this environment variable to take effect.

Copy and Paste Troubleshooting

For Windows applications and X applications, users can only copy and paste *text* under the following conditions:

- In the Administration Console, go to the Global Settings → Client Device tab, copy and paste for SGD as a whole must be enabled. Copy and paste is enabled by default.
- The user must be allowed to copy and paste. If the Copy and Paste attribute on the Client Device tab for the user profile is selected, then the user can copy and paste. This attribute might be configured to use the setting of any parent organizational unit or organization object. Copy and paste is enabled by default.
- To be able to paste data to another Windows application or X application displayed through SGD, the *source application* must have an Application's Clipboard Security Level that is lower than, or equal to, the *target application*. The source application is the application the data is copied from. The target application is the application the data is pasted to. The default security level is 3.
- To be able to paste data to an application running on the client device, the source application must have an Application's Clipboard Security Level that is lower than, or equal to, the Client's Clipboard Security Level. The Client's Clipboard Security Level is shown on the Global Settings → Client Device tab of the Administration Console. The default Client's Clipboard Security Level is 3.

If these conditions are not met, users paste the following message, instead of the copied data: Oracle Secure Global Desktop Software: Copied data not available to this application

For Windows applications, users you can only copy *graphics* from, or paste graphics to, Microsoft Windows 2000 or later applications.

To copy and paste Unicode text in X applications, the X application must support Unicode. Common Desktop Environment (CDE) and Motif applications, for example, do not support Unicode.

Smart Cards

This section describes how to configure smart cards for Windows applications displayed through SGD.

This section includes the following topics:

- “Using Smart Cards With Windows Applications” on page 292
- “Setting Up Access to Smart Cards” on page 293
- “Configuring the Microsoft Windows Application Server for Smart Cards” on page 293
- “Enabling Smart Cards in SGD” on page 294
- “Configuring Smart Card Readers on Client Devices” on page 295
- “How to Log In to a Microsoft Windows Application Server With a Smart Card” on page 296
- “Troubleshooting Smart Cards” on page 296

Using Smart Cards With Windows Applications

SGD enables users to access a smart card reader attached to their client device from applications running on a Windows application server. Users can do the following:

- Use a smart card to log in to a Windows application server.
- Access the data on a smart card while using an application running on a Windows application server. For example, to use a certificate for signing or encrypting an email.

SGD works with any Personal Computer/Smart Card (PC/SC)-compliant smart card and reader. Details of the smart cards that have been tested successfully with SGD are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

Setting Up Access to Smart Cards

SGD Administrators can give users access to smart card readers from Windows applications displayed through SGD. Setting up access to smart cards involves the following configuration steps:

1. Enable smart card services on the application server.
See [“Configuring the Microsoft Windows Application Server for Smart Cards”](#) on page 293.
2. Enable access to smart cards for SGD users.
See [“Enabling Smart Cards in SGD”](#) on page 294.
3. Configure a smart card reader on the client device.
See [“Configuring Smart Card Readers on Client Devices”](#) on page 295.
4. Log in to the application server using the smart card.
See [“How to Log In to a Microsoft Windows Application Server With a Smart Card”](#) on page 296.

Configuring the Microsoft Windows Application Server for Smart Cards

To configure the Microsoft Windows application server for smart cards, do the following:

- Deploy smart cards on the Microsoft Windows Server domain.
See [Planning a Smart Card Deployment](#) for the main configuration steps involved when deploying smart cards.
- Check that smart card device redirection is enabled on the Windows Terminal Server. See [“Configuring Microsoft Windows Terminal Services for Use With SGD”](#) on page 140 for details of the Windows platforms that support smart card device redirection.
- Ensure that smart cards are working before introducing SGD.

Application Server Authentication Dialog Settings

In the Administration Console, the Global Settings → Application Authentication tab has several attributes that control the behavior of the Application Server Authentication dialog when using the SGD smart card service.

The Smart Card Authentication check box controls whether users get the choice of logging in with a smart card or only with a user name and password.

The "Always Use Smart Card" Box attributes enable you to control whether a user's decision to log in with a smart card is remembered, or *cached*, for the next time they log in to that application server, and whether they can change this setting.

Note – Users can only choose an authentication method, or to cache the smart card decision, if they have access to the Application Server Authentication dialog. If you disable the ability to use Shift-click, this restricts user access to the Application Server Authentication dialog. See [“Users Can Start Applications With Different User Names and Passwords” on page 212.](#)

Enabling Smart Cards in SGD

SGD must be configured in order to support user access to smart cards.

Firewalls between SGD servers can interfere with the connections required for smart cards, see [“Firewalls Between SGD Servers” on page 14.](#)

▼ How to Enable Smart Cards in SGD

1. Check that the SGD smart card service is enabled.

In the Administration Console, go to the Global Settings → Client Device tab, ensure the Smart Card check box is selected.

The smart card service is enabled by default.

2. Ensure that smart card authentication is enabled.

Smart card authentication is enabled by default.

In the Administration Console, go to the Global Settings → Application Authentication tab, ensure the Smart Card Authentication check box is selected.

The Global Settings → Application Authentication tab has other settings that affect the behavior of the Always Use Smart Card check box on the Application Server Authentication dialog. See [“Application Server Authentication Dialog Settings” on page 293.](#)

Configuring Smart Card Readers on Client Devices

SGD works with PC/SC-compliant cards and readers. See the [PC/SC Workgroup web site](#) for more information.

The smart cards tested with SGD are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

Microsoft Windows Client Devices

On Microsoft Windows client devices, you must install the smart card reader and any required drivers on the client device to make the smart card available to Terminal Services sessions running through SGD.

Linux Platform and Solaris OS Client Devices

On Linux platform and Solaris OS client devices, a PCSC-Lite library must be installed for SGD to communicate with smart card readers. PCSC-Lite provides an interface to the PC/SC framework on UNIX and Linux platforms.

For Linux platform client devices, PCSC-Lite is available from the following locations:

- Your Linux platform vendor.
- The [MUSCLE project](#).

PCSC-Lite version 1.2.0 or later is required.

For Solaris OS client devices, PCSC-Lite compatible libraries are available in the following packages:

- The PC/SC Shim for SCF package (PCSCshim)
- The Sun Ray PC/SC Bypass package (SUNWsrcbp)

The PC/SC Shim for SCF package enables you to use a PC/SC application with the Solaris Card Framework (SCF) and work with Sun internal readers and Sun Ray readers. Version 1.1.1 or later is required. PC/SC Shim is included with Solaris 10. For other Solaris versions, PC/SC Shim is available from the [MUSCLE project](#).

The Sun Ray PC/SC Bypass package provides a PCSC-Lite interface for the Ray reader. Make sure you have the latest patches for Sun Ray Server Software and the latest SUNWsrcbp package.

SGD clients require the PCSC-Lite `libpcsc-lite.so` library file. This is normally installed in `/usr/lib`, but the location depends on your dynamic linker path. If this file is installed outside of the dynamic linker path, or you want to use a different library file, use the `TTA_LIB_PCSC-LITE` environment variable to specify the location. This can be set either in the user's environment or in the login script.

▼ How to Log In to a Microsoft Windows Application Server With a Smart Card

1. Log in to SGD.
2. On the webtop, click the link to start the Windows application.
3. When the Application Server Authentication dialog displays, click Use smart card.
4. To always use a smart card to log in, click the Always use smart card box.
5. When the Windows security dialog displays, insert your smart card.
6. When prompted, enter your PIN.

Troubleshooting Smart Cards

For information about configuring SGD to use smart cards with Windows applications see [“Using Smart Cards With Windows Applications” on page 292](#).

If users find they are unable to use their smart cards with Windows applications, use the following checklist to resolve the problem.

Is the smart card device redirection enabled on the Windows Terminal Server?

You can only use smart cards if smart card device redirection is enabled on the Windows Terminal Server. See [“Configuring Microsoft Windows Terminal Services for Use With SGD” on page 140](#) for details of the Windows platforms that support smart card device redirection.

Are smart card services enabled for all SGD servers in the array?

In the Administration Console, go to the Global Settings → Client Device tab, ensure the Smart Card check box is selected.

In the Administration Console, go to the Global Settings → Application Authentication tab, ensure the Smart Card Authentication check box is selected.

Is there a firewall between the SGD server hosting the user session and the SGD server hosting the application session?

Firewalls between SGD servers can interfere with smart card connections, see [“Firewalls Between SGD Servers”](#) on page 14.

Is the client device configured correctly?

On *Microsoft Windows* client platforms, do the following:

- Check that the smart card reader is listed in the Windows Device Manager.
- Check that the smart card service is running on the client. Click Start Menu → Programs → Administrative Tools → Services.
- Check that the SGD Client has detected the smart card reader and card. Click the right mouse button on the SGD icon in the Windows system tray and select Connection info. The Smart card reader property lists the details in the format *reader:ATR_string* where *reader* is the manufacturer and model of the smart card reader and *ATR_string* is the Automatic Terminal Recognition (ATR) string, a sequence of hexadecimal numbers used to identify the card to the system.

On *Linux* platforms, do the following:

- Check that the PCSC daemon, `pcscd`, is running. For example, you can use the following command:

```
# /sbin/service pcscd status
```

- Try restarting the PCSC daemon with a `--debug stdout` option. Insert the smart card in the reader and see if the reader and card are detected.

On *Solaris OS* platforms, do the following:

- If you are using the PC/SC Shim for SCF package, check that the OCF server, `ocfserv`, is running. If the OCF server is not running, use the following command to enable the OCF server:

```
# svcadm enable svc:/network/rpc/ocfserv
```

- If you are using the Sun Ray PC/SC Bypass package, check the Sun Ray Server Software configuration.

Are there any error messages listed in the log file?

Smart card device access data and error messages are stored in the SGD Client log file. This data is displayed in the Detailed Diagnostics page of the SGD webtop.

Serial Ports

This section describes how to set up access to serial ports for Windows applications displayed through SGD.

This section includes the following topics:

- [“Setting Up Access to Serial Ports” on page 298](#)
- [“Configuring the Microsoft Windows Application Server” on page 298](#)
- [“Enabling Serial Port Access in SGD” on page 298](#)
- [“Configuring the Client Device” on page 299](#)

Setting Up Access to Serial Ports

Setting up access to serial ports involves the following configuration steps:

1. Enable COM port mapping on the application server.
See [“Configuring the Microsoft Windows Application Server” on page 298](#).
2. Enable access to serial ports for SGD users.
See [“Enabling Serial Port Access in SGD” on page 298](#).
3. Configure the client device for serial port access.
See [“Configuring the Client Device” on page 299](#).

Configuring the Microsoft Windows Application Server

You can only access serial ports if COM port mapping is enabled on the Windows Terminal Server. See [“Configuring Microsoft Windows Terminal Services for Use With SGD” on page 140](#) for details of the Windows platforms that support COM port mapping.

Enabling Serial Port Access in SGD

Access to serial ports is enabled for all users by default. If serial port access is disabled, you can enable access to serial ports for all users, or for specific users.

When a user starts a Windows application, SGD checks the user profile for the user and then any parent object further up the organizational hierarchy to see whether access to serial ports is enabled or disabled. If all the objects checked are configured to use the parent's setting, then the global setting is used.

Firewalls between SGD servers can interfere with the connections required for serial ports, see [“Firewalls Between SGD Servers” on page 14](#).

▼ How to Enable Access to Serial Ports

1. **In the Administration Console, go to the Global Settings → Client Device tab and select the Serial Port Mapping check box.**

The Serial Port mapping check box is enabled by default.

2. **(Optional) In the Administration Console, go to the Client Device tab for an organization, an organizational unit, or a user profile object.**

- a. **Select the Override Parent's Settings or Override Global Settings check box.**

- b. **Set the Serial Port Mapping attribute.**

To enable access to serial ports, select the Enabled check box. To disable access to serial ports, deselect the Enabled check box.

If you configure an organization or organizational unit object, this affects all the users in that organization or organizational unit.

Note – The changes made only take effect for new user sessions.

Configuring the Client Device

To determine the serial ports that are mapped in the Windows Terminal Services session, you might have to configure the client device.

On *UNIX* and *Linux* client platforms, users must have read and write access to any serial device that is mapped. SGD uses the *first match* of the following:

1. The serial ports listed in the `SUN_MAP_SERIALPORTS` environment variable.

Each serial port in the list is separated with a semi-colon and has the format `serial device=com-port-name`. For example:

```
/dev/ttyS0=COM1 ; /dev/ttyS4=COM8
```

The `=com-port-name` part is optional, but if it is omitted the serial port is mapped to COM x in the Windows application session, where x is the position of the serial port in the list.

2. The serial ports listed in the user's client profile.

The `<serialports>` entry in the `<localsettings>` section of the user's client profile lists the serial ports to be mapped. See ["Client Profile Settings" on page 315](#).

The `<serialports>` entry has to be added manually.

The serial ports are listed in the same format as above.



Caution – If a user has not edited their client profile, any manual changes made to the `profile.xml` file are lost when the user next logs in.

3. The serial port listed in the `SUN_DEV_SERIAL` environment variable.

This is a single serial device, for example `/dev/ttyS2`. This is always mapped to COM1 in the Windows application session.

On *Microsoft Windows* client platforms, SGD uses the *first match* of the following:

1. The serial ports listed in the user's client profile.

The `<serialports>` entry in the `<localsettings>` section of the user's client profile lists the serial ports to be mapped. See ["Client Profile Settings" on page 315](#).

The `<serialports>` entry has to be added manually.

Each serial port in the list is separated with a semi-colon and has the format `serial device=com-port-name`.

`COM1=COM5;COM2=COM8`

The `=com-port-name` part is optional, but if it is omitted the serial port is mapped to COM x in the Windows application session where x is the position of the serial port in the list.



Caution – If a user has not edited their client profile, any manual changes made to the `profile.xml` file are lost when the user next logs in.

2. Any available COM1 to COM9 ports.

The SGD Client attempts to open ports COM1 to COM9. If a COM port is found, it is mapped to the same COM port number in the Windows application session.

SGD Client and Webtop

This chapter describes how to install, configure, and run the Oracle Secure Global Desktop (SGD) Client. Webtop configuration is also covered.

This chapter includes the following topics:

- [“The SGD Client” on page 301](#)
- [“Client Profiles” on page 311](#)
- [“Integrated Mode” on page 320](#)
- [“Webtops” on page 329](#)

The SGD Client

The SGD Client is the part of SGD that is installed on client devices. The SGD Client is required to run applications.

This section includes details of how you can install and run the SGD Client.

This section includes the following topics:

- [“Overview of the SGD Client” on page 301](#)
- [“Installing the SGD Client” on page 303](#)
- [“Running the SGD Client From the Command Line” on page 306](#)
- [“Using SGD Without Java Technology” on page 309](#)

Overview of the SGD Client

The SGD Client can operate in either of the following ways:

- **Using a browser.** You can use a browser to display a special web page, called a *webtop*. The webtop lists the applications a user can run through SGD and provides controls for managing application sessions and printing. This is the default way of using SGD.

See the *Oracle Secure Global Desktop 4.6 User Guide* for more details about the webtop.

- **Integrated mode.** The list of applications that a user can run through SGD is shown in the desktop Start or Launch Menu on the client device. Using Integrated mode enables users to run remote applications in the same way as local applications. Depending on other configuration factors, users might not need to use a browser.

See [“Integrated Mode” on page 320](#) for more details.

Depending on the client platform, users see an icon in the System tray or Workspace switcher when the SGD Client is running.

The SGD Client performs the following functions:

- Gets information about the client device, such as the operating system, local printers, and client drives.
- Manages the display of applications.
- Maintains a communication connection with the SGD server, using the Adaptive Internet Protocol (AIP) protocol.
- Receives and acts on events from the SGD server. For example, the arrival of a print job.

Configuring the SGD Client

The SGD Client needs to be configured so that it can connect to an SGD server. The connection settings for the SGD Client are defined in a *client profile*. The client profile is stored on the client device.

The client profile controls things such as the Uniform Resource Locator (URL) that the SGD Client connects to when it starts, and the operating mode of the SGD Client.

See [“Client Profiles” on page 311](#) for more information about how SGD uses client profiles and the settings you can configure for a client profile.

The SGD Client Helper

When using a browser with Java technology enabled, the SGD Client is supported by the *SGD Client Helper*.

The SGD Client Helper is a Java applet that performs the following functions:

- Downloads and installs the SGD Client. This only applies if automatic installation is used. See also [“Automatic Installation of the SGD Client” on page 303](#).
- Obtains proxy server settings from the browser and sends them to the SGD Client. This depends on the settings in the user’s client profile.
- Starts the SGD Client. This only happens when a user starts a browser and goes to the login URL.
- Responds to instructions received from the SGD Client. For example, prompting the browser to redraw the screen.

Use of the SGD Client Helper is optional. See [“How to Use SGD Without Java Technology” on page 310](#).

Installing the SGD Client

The SGD Client can be installed in the following ways:

- **Automatic installation.** Download and installation of the SGD Client can be handled automatically, using a browser with Java technology enabled. See [“Automatic Installation of the SGD Client” on page 303](#).
- **Manual installation.** The SGD Client can be downloaded to the client device and installed manually. See [“Manual Installation of the SGD Client” on page 305](#).

Automatic Installation of the SGD Client

If you are using a browser with Java technology enabled, the SGD Client is installed automatically when you visit the `http://server.example.com/sgd` URL, where *server.example.com* is the name of an SGD server.

Note – If you use Internet Explorer on Microsoft Windows Vista platforms, you must add the SGD server to the list of Trusted Sites in Internet Explorer’s Security Settings before the SGD Client can be automatically downloaded and installed.

With automatic installation of the SGD Client, different versions of the SGD Client are installed in separate directories. This means the following:

- Users only have to log in to an upgraded SGD server in order to upgrade the SGD Client
- Users who log in to different SGD servers always run the correct SGD Client for the version of SGD

The SGD Client is installed in the following directories:

- **Microsoft Windows client devices.** A user-specific writeable directory.

On Microsoft Windows XP platforms, for example:

```
C:\Documents and Settings\username\Local Settings\Temp\tcc\version
```

On Microsoft Windows Vista platforms, for example:

```
C:\Users\username\AppData\Local\Temp\tcc\version
```

The actual location depends on the user's privileges, the operating system, and the version of the Java Plugin tool being used.

Users with Microsoft Windows client devices can have *roaming user profiles*. Roaming user profiles provide the user with the same working environment, no matter which Microsoft Windows computer they use.

If Microsoft Windows users have roaming user profiles, the SGD Client is installed in one of the following directories:

On Microsoft Windows XP platforms:

```
C:\Documents and Settings\username\Application Data\Temp\tcc\version
```

On Microsoft Windows Vista platforms:

```
C:\Users\username\AppData\Roaming\Temp\tcc\version
```

For details of how to configure SGD to work with roaming user profiles, see [“How to Enable Automatic Installation for Roaming User Profiles”](#) on page 304.

- **UNIX®, Linux, or Mac OS X platform client devices.** The user's home directory:
`$HOME/.tarantella/tcc/version`

If you want to use automatic installation and have more control over where the SGD Client is installed, you can develop your own web application for installing the SGD Client and use SGD web services to specify the installation location.

See the *Oracle Secure Global Desktop 4.6 Installation Guide* for more details about automatic installation of the SGD Client.

▼ How to Enable Automatic Installation for Roaming User Profiles

To enable the SGD Client to be installed automatically in a directory that is roamed, perform the following procedure on each SGD server in the array.

Ensure that no users are logged in to the SGD server, and that there are no application sessions, including suspended application sessions, running on the SGD server.

1. **Log in as superuser (root) on the SGD host.**

2. Change to the jsp directory.

```
# cd /opt/tarantella/webserver/tomcat/tomcat-version/webapps/sgd/resources/jsp
```

3. Edit the webtopsession.jsp file.

Change the tccRoaming line in webtopsession.jsp, as follows:

```
String tccRoaming = "true";
```

Save the change.

4. Restart the SGD web server.

```
# tarantella restart webserver
```

Manual Installation of the SGD Client

With manual installation, you have full control over where the SGD Client is installed.

On Microsoft Windows platforms, you need administrator privileges to install the SGD Client.

You download and install the SGD Client from the SGD web server Welcome Page. The SGD web server Welcome Page is at <http://server.example.com>, where *server.example.com* is the name of an SGD server.

Click the Install the Oracle Secure Global Desktop Client link on the Welcome Page. The Oracle Secure Global Desktop Client download page has instructions for downloading and installing the SGD Client.

On Microsoft Windows client devices, the default installation directory is: C:\Program Files\Sun\Secure Global Desktop Client. A shortcut for the SGD Client is also added to the Windows Start Menu.

Note – Manual installation is not available for Mac OS X client platforms.

See the *Oracle Secure Global Desktop 4.6 Installation Guide* for more details about manual installation of the SGD Client.

Running the SGD Client From the Command Line

Typically, users log in to SGD by starting a browser and visiting the `http://server.example.com/sgd` URL, where `server.example.com` is the name of an SGD server.

Connecting to SGD in this way, automatically downloads and starts the SGD Client. However, you can also start the SGD Client from the command line and connect to an SGD server. From the command line, you can run the SGD Client either using a browser or in Integrated mode.

You start the SGD Client with the `tcc` command on Microsoft Windows client platforms, or the `ttatcc` command on UNIX, Linux, or Mac OS X client platforms, as follows:

```
tcc
[ -profile name ]
[ -loginurl url ]
[ -preferredlanguage lang ]
[ -logdir file ]
[ -use-java ]
[ -version ]
```

The following table lists the arguments for the `tcc` and `ttatcc` commands.

| Argument | Description |
|---|--|
| <code>-profile <i>name</i></code> | The name of the profile to use when starting the SGD Client. Currently there is only one profile for each SGD server, called Default. To specify the profile for a particular server, use <code>-profile server.example.com::Default</code> where <code>server.example.com</code> is the name of an SGD server. Note - Profile names are case sensitive. |
| <code>-loginurl <i>URL</i></code> | The login URL. This overrides the URL defined in the profile. Use a fully qualified domain name. |
| <code>-preferredlanguage <i>lang</i></code> | The language to use in any dialogs and messages displayed by the SGD Client. This overrides the language defined in the profile. The following are the supported languages: <ul style="list-style-type: none">• <code>en</code> for English• <code>de</code> for German• <code>fr</code> for French• <code>ja</code> for Japanese• <code>ko</code> for Korean• <code>zh_CN</code> for Simplified Chinese• <code>zh_TW</code> for Traditional Chinese |

| Argument | Description |
|---------------------------|--|
| <code>-logdir file</code> | The directory where the SGD Client log file is created. |
| <code>-use-java</code> | Enable the detection of Java technology in the SGD Client. |
| <code>-version</code> | Displays the version number of the SGD Client. |
| <code>-help</code> | Displays help information. This option is only available on UNIX, Linux, or Mac OS X client platforms. |

Note – The arguments are case-sensitive.

The command line does not allow you to supply a user name and password. However, the SGD Client can be configured to log a user in automatically. This is called *Integrated mode*. See “Setting Up the SGD Client for Integrated Mode” on page 322 for more details.

Command-Line Examples

The command line for the SGD Client can be used to create your own shortcuts and shell scripts.

If either the Connect on System Login or the Add Applications to Start Menu options are enabled in a user’s profile, the SGD Client automatically adds shortcuts for itself in the user’s desktop Start menu. The desktop systems that support these features are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

The following are some examples of running the SGD Client from the command line.

Starting the SGD Client Without Any Arguments

The following example starts the SGD Client and uses the settings defined in the Default profile, available from the user’s profile cache.

```
$ ttatcc
```

If there is no profile, or the profile does not contain a login URL, the SGD Client starts but it cannot connect to an SGD server.

If the user has previously connected to more than one SGD server, the SGD Client connects to the last SGD server the user connected to, using the profile for that server.

Use this command to start the SGD Client if the user always connects to the same SGD server.

Connecting to a Particular SGD Server

The following example starts the SGD Client and uses the settings defined in the profile for *server.example.com*, available from the user's profile cache.

```
$ ttatcc -profile server.example.com::Default
```

If there is no profile available in the cache for *server.example.com*, the SGD Client prompts for connection settings.

Use this command to start the SGD Client if the user might connect to different SGD servers.

Overriding the Login URL

The following example starts the SGD Client and uses the settings defined in the Default profile, available from the user's profile cache, but connects to the specified URL.

```
$ tcc -loginurl url
```

where *url* is the URL of a login page on an SGD server, for example `http://server.example.com/sgd`.

Depending on the URL, this can be used to start an application.

Use this command to start the SGD Client and connect to a single SGD server, but connect to different web applications on that server.

Web Services Developer Options

The SGD Client also supports the following command-line arguments. These arguments are useful only when developing applications with SGD web services.

| Argument | Description |
|--|---|
| <code>-port <i>tcp</i></code> | The port on which the SGD Client connects to the SGD server. Usually, this is Transmission Control Protocol (TCP) port 5307 when the user has a secure connection to SGD. |
| <code>-baseroute</code> | The base network route the SGD Client uses to traverse a SOCKS proxy server. |
| <code>-firewalltraversal</code> | Indicates that the SGD server is using firewall traversal. Connections to the SGD server and the webtop both use the same port, usually TCP port 443. |
| <code>-connectioncookie <i>cookie</i></code> | Supplies the cookie used by the SGD server to identify the user session which the SGD Client is being used for. |
| <code>-portfile <i>file</i></code> | The name of a file where the SGD Client writes its listening port number. |
| <code>-psn</code> | For use with Mac OS X client devices only. Ensures an X server is running. |
| <code>-server <i>server</i></code> | The fully-qualified Domain Name System (DNS) name of the SGD server. |
| <code>-no-browser</code> | Do not start a browser when starting the SGD Client. |

Note – The arguments are case-sensitive.

Using SGD Without Java Technology

If your organization prefers not to use Java technology, you can still use SGD, but with the following limitations:

- The SGD Client must be manually downloaded and installed.
- To log in to SGD, users must start the SGD Client *first*, either from the command line or from the shortcut in the desktop Start menu. Users cannot log in by starting a browser first.
- Proxy server settings must be configured users' client profiles. The settings cannot be obtained automatically from a browser.
- The webtop must be reloaded manually to show the current state.

The following procedure describes the steps needed to use SGD without Java.

▼ How to Use SGD Without Java Technology

1. Download and install the SGD Client.

You download the SGD Client from the SGD web server Welcome Page, for example at `http://server.example.com`, where *server.example.com* is the name of an SGD server.

Click the link to Install the Oracle Secure Global Desktop Client.

The download page and the *Oracle Secure Global Desktop 4.6 Installation Guide* have details of how to install the SGD Client.

2. Start the SGD Client and connect to SGD.

Use either of the following methods:

- Start the SGD Client from the shortcut in the desktop Start menu.

The first time you start the SGD Client, it prompts you for the URL to connect to. This is normally `http://server.example.com/sgd`, where *server.example.com* is the name of an SGD server. The SGD Client also prompts you for the proxy server settings to use.

When the SGD Client connects, it starts your default browser and displays the SGD login page.

- Start the SGD Client from the command line.

See [“Running the SGD Client From the Command Line” on page 306](#) for more details.

3. Log in to SGD.

The SGD webtop is displayed.

4. Edit the profile for your client device.

On the webtop, click the Edit button in the Applications area of the webtop. Go to the Client Settings tab and edit the client profile.

See also [“Client Profile Settings” on page 315](#).

a. Configure the operating mode of the SGD Client.

You can access SGD either by using a browser or by using Integrated mode.

Integrated mode gives users the best user experience when Java technology is unavailable. Select the Add Applications to Start Menu check box. See also [“Integrated Mode” on page 320](#).

To use automatic logins to minimize the use of a browser, select the Automatic Client Login check box. See [“Authentication Token Authentication” on page 323](#).

Whenever the SGD Client needs to display a page in a browser, for example to display a webtop or a login page, it always starts the *default* browser.

To update the webtop display, users might have to manually reload the page.

b. Configure the proxy server settings.

You must specify the proxy server settings in the profile, because these settings cannot be obtained from the browser. See [“Configuring Client Proxy Settings” on page 9](#).

c. Click Save.

Note – SGD Administrators can preconfigure many of these settings for users, by editing the profile for an organization or organizational unit.

5. Log out of SGD.

Client Profiles

This section includes details on how to manage and configure client profiles for the SGD Client.

This section includes the following topics:

- [“Client Profiles and the SGD Client” on page 312](#)
- [“Managing Client Profiles” on page 312](#)
- [“Client Profile Settings” on page 315](#)
- [“About the Profile Cache” on page 317](#)
- [“Microsoft Windows Users With Roaming User Profiles” on page 319](#)

Client Profiles and the SGD Client

A *client profile* is a group of configuration settings that control the SGD Client. The settings in a client profile include the following:

- The URL the SGD Client connects to when it starts. Usually, this is the URL used to log in to SGD.
- The operating mode of the SGD Client. Whether the applications a user can run are displayed on a webtop or in the user's desktop Start or Launch Menu.
- Whether or not the user is logged in automatically to SGD when the SGD Client starts.
- Whether or not the SGD Client starts automatically when the user logs in to their desktop system.
- Proxy server configuration. Whether the proxy settings are manually configured in the profile or determined from the browser.

Note – The SGD Client can only connect to an SGD server if they both have the same major and patch version number. For example, version 4.40.917.

There is one client profile, a single group of settings, for each SGD server that the user connects to. The profile is downloaded when the user connects to an SGD server. If the SGD Client has been installed manually, the user is prompted for initial connection information the first time the SGD Client is started.

Note – Client profiles are not the same as user profiles. User profiles control webtop content and other SGD-specific settings, such as printing.

This section includes the following topics:

- [“Managing Client Profiles” on page 312](#)
- [“How to Configure Client Profile Editing for Users” on page 313](#)
- [“Client Profile Settings” on page 315](#)
- [“About the Profile Cache” on page 317](#)
- [“Microsoft Windows Users With Roaming User Profiles” on page 319](#)

Managing Client Profiles

SGD Administrators manage client profiles with the SGD administration tool, Profile Editor. The Profile Editor tool is only available to SGD Administrators.

SGD Administrators can create, edit, and delete client profiles for the following objects:

- Organization objects
- Organizational unit (OU) objects
- Profile objects in the `System Objects` organization. For example, `System Objects/LDAP Profile`

Each of these objects can only have one client profile. The client profile is stored on the SGD server.

The default system client profile is the profile for the `System Objects` organization. This client profile can be edited, but it cannot be deleted.

Users can edit their own client profiles from the webtop. Click the Edit button in the Applications area of the webtop and then go to the Client Settings tab.

Users can only edit the client profile for the SGD server they are currently connected to. The client profile for a user is stored on the client device, not the SGD server.

Note – Anonymous users cannot edit client profiles. This is because these users are temporary. See [“Anonymous User Authentication” on page 61](#) for more details.

▼ How to Configure Client Profile Editing for Users

1. Enable profile editing for SGD.

Profile editing for SGD is enabled by default.

a. **In the Administration Console, go to the Global Settings → Client Device tab.**

b. **In the Profile Editing section, ensure the Editing check box is selected.**

The check box is selected by default.

Note – If profile editing is disabled, it is disabled for *all* users, including SGD Administrators. However, SGD Administrators can still create and edit client profiles using the Profile Editor application.

2. Configure profile editing in the organizational hierarchy.

Profile editing can be configured for organizations, organizational units, or user profiles.

Profile editing can be inherited from a parent object in the organizational hierarchy, so that SGD Administrators can enable or disable profile editing for many users without having to edit each user profile. By default, profile editing is enabled for all users.

a. In the Administration Console, go to the User Profiles tab and select an object in the organizational hierarchy.

b. Go to the Client Device tab.

c. Enable Client Profile Editing as follows:

- Select the Override Parent's Setting, or the Override Global Setting check box.

Selecting this check box enables you to override the profile editing setting from any parent object. For example, profile editing can be disabled for an OU, but enabled for a user profile in that OU.

- Select the Enabled check box.

Selecting the check box enables profile editing for the user profile, or for all users in the organization unit or organization.

The initial state of this check box is the setting of the parent object.

d. Click Save.

Client Profile Settings

The following table lists the settings available in a client profile, with a description of what the setting does.

| Setting | Description |
|--------------------------------|--|
| Login URL | <p>The SGD URL to use for the profile. This is usually <code>http://server.example.com/sgd</code>, where <i>server.example.com</i> is the name of an SGD server.</p> <p>If the user runs SGD by displaying the webtop in a browser, the URL is loaded automatically in the user's default browser, so that they can log in and access their webtop.</p> <p>In Integrated mode, the URL is only loaded in the user's default browser if the user needs to log in to SGD.</p> <p>Always use a fully qualified domain name.</p> <p>The URL in a client profile can be overridden by a command-line argument. See "Running the SGD Client From the Command Line" on page 306.</p> <p>The default Login URL is <code>http://server.example.com:80/sgd/index.jsp</code>.</p> |
| Connect on System Login | <p>If enabled, the SGD Client is started automatically with this client profile whenever the user logs in to their client device.</p> <p>If enabled, the SGD Client creates an application shortcut or symbolic link for itself in the startup folder of the desktop system. The links are created in the following locations:</p> <ul style="list-style-type: none">• Microsoft Windows. The Windows startup folder for the current user. This is usually <code>C:\Documents and Settings\username\Start Menu\Programs\Startup</code>• KDE. <code>\$HOME/.kde/autostart</code>• Gnome. <code>\$HOME/.config/autostart</code>• Java Desktop System. <code>\$HOME/.config/autostart</code> <p>This setting is disabled by default.</p> |
| Add Applications to Start Menu | <p>Controls how users interact with SGD.</p> <p>If enabled, the applications a user can run are displayed in the desktop Start or Launch Menu on the client device. This is called Integrated mode. Users do not have any of the controls that are available on a webtop, for example controls for suspending and resuming applications.</p> <p>If disabled, the applications a user can run are displayed on a webtop in a browser.</p> <p>This setting is disabled by default.</p> |
| Automatic Client Login | <p>If enabled, the SGD Client tries to log the user in using an authentication token as soon as it starts.</p> <p>You can only enable this option if the Add Applications to Start Menu setting is enabled.</p> <p>This setting is disabled by default.</p> <p>See "Integrated Mode" on page 320 for more details.</p> |

| Setting | Description |
|------------------------|---|
| Alternative PDF Viewer | <p>The application command for an alternative Portable Document Format (PDF) viewer to use with PDF printing.</p> <p>If the application is not on the user's <code>PATH</code>, type the full path to the application.</p> <p>This setting only applies to UNIX, Linux, and Mac OS X platform client devices.</p> |
| Logging | <p>Controls the amount of information that is output to the SGD Client log file.</p> <p>The output is logged to a text file in the same directory as the SGD Client.</p> <p>The default is Errors only.</p> |
| Preferred Language | <p>The default language to use when the SGD Client is started from the command line. For example, when the SGD Client is in Integrated mode.</p> <p>The language selected is used for messages displayed by the SGD Client, the login dialog, and the webtop.</p> <p>See “Setting the Language for the Webtop” on page 329 for details.</p> <p>The default is en.</p> |

| Setting | Description |
|--------------------------|--|
| Check for Local X Server | <p>If enabled, the SGD Client checks whether there is an X server running on the client device.</p> <p>Enabling this option can improve performance when starting X applications that are configured to display using an X server on the client device. If a local X server is not available, an independent window is used instead.</p> <p>This setting only applies to Windows client devices.</p> <p>This setting is disabled by default.</p> |
| Proxy Settings | <p>Settings that control how the SGD Client determines what proxy servers to use.</p> <p>Use Default Web Browser Settings means use the proxy server settings configured in the user's default browser.</p> <p>Manual Proxy Settings enable you to define the proxy server settings in the profile. You can specify an Hypertext Transfer Protocol (HTTP) proxy server. If the proxy settings are determined from a browser, the settings are stored and used the next time the SGD Client starts.</p> <p>If Establish Proxy Settings on Session Start is enabled, the SGD Client obtains the proxy settings from the browser every time it starts. The stored proxy settings are not used. If Automatic Client Login is selected, the Establish Proxy Settings on Session Start setting is disabled.</p> <p>By default, the Use Default Web Browser Settings check box is selected and the Establish Proxy Settings on Session Start check box is not selected.</p> |
| Connection Failure | <p>Settings that control what the SGD Client does if the connection to an SGD server is lost, whether to always reconnect, to never reconnect, or to ask the user.</p> <p>If the SGD Client reconnects, these settings control how many attempts are made to reconnect and the time in seconds between each attempt.</p> <p>If the SGD Client is unable to reconnect, the user session ends and any running applications are ended or suspended, depending on the resumability setting of the application.</p> <p>The default settings are to Always Attempt to Reconnect, and make 6 attempts at 10 second intervals.</p> |

About the Profile Cache

Client profiles created by SGD Administrators are stored on the SGD server where they are created. The profiles are then copied to all the SGD servers in the array, so that they are available for editing on any SGD server.

When a user first logs in to SGD, the SGD Client downloads the client profile to a profile cache on the client device. The client profile that is downloaded is the first match of the following:

- The client profile defined for a user profile object in the System Objects organization that is assigned to the user. For example, if the user is authenticated using LDAP authentication and a client profile exists for the System Objects/LDAP Profile object, this is the profile that is downloaded.
- The client profile defined by an SGD Administrator for the organizational unit or organization to which the user belongs. If there is no client profile for the user's organizational unit, SGD checks any parent object further up the organizational hierarchy to see whether they have a client profile.
- The system default client profile defined for the System Objects object.

When a user edits and saves a client profile, they override the client profile defined by an SGD Administrator, or the system default client profile, and create a user-specific client profile that is only saved in the profile cache on the client device.

Note – Users must log out of SGD and log in again for changes to their client profile to take effect.

The profile cache is specific to each user who logs in to SGD from the client device and is stored in the following locations:

- **UNIX, Linux, and Mac OS X platform client devices** –
\$HOME/.tarantella/tcc/profile.xml
- **Microsoft Windows XP client devices** – C:\Documents and Settings\
username\Local Settings\Application Data\Sun\SSGD\profile.xml
- **Microsoft Windows Vista client devices** – C:\Users\username\AppData\
Local\Sun\SSGD\profile.xml

Note – If a Windows user has a roaming user profile, see [“How to Enable Automatic Installation for Roaming User Profiles”](#) on page 304.

The same profile cache is used by the SGD Client, whether it has been installed manually or automatically.

The profile cache is updated each time the user edits a client profile, or each time the user logs in, if they are using the client profile defined by an Administrator.



Caution – If a user has not edited their client profile, any manual changes made to the profile.xml file are lost when the user next logs in.

The profile cache contains one client profile for each SGD server the user connects to.

Users can restore a client profile to the default settings by editing the client profile and clicking the Reset button. This resets the client profile to the settings defined for the system default client profile on the `System Objects` object.

Microsoft Windows Users With Roaming User Profiles

Users with Microsoft Windows client devices can have *roaming user profiles*. Roaming user profiles provide the user with the same working environment, no matter which Microsoft Windows computer they use. If Microsoft Windows users have roaming user profiles, the SGD client profile is automatically adjusted to allow for this, as follows:

- Settings specific to the user's client device, for example the proxy server configuration, are stored on the client device in the following default locations:
 - **Microsoft Windows XP client devices** – `C:\Documents and Settings\username\Local Settings\Application Data\Sun\SSGD\profile.xml`
 - **Microsoft Windows Vista client devices** – `C:\Users\username\AppData\Local\Sun\SSGD\profile.xml`
- Settings specific to the user, for example the preferred language, are stored in the location of the roaming user profile:
 - **Microsoft Windows XP client devices** – `C:\Documents and Settings\username\Application Data\Sun\SSGD\profile.xml`
 - **Microsoft Windows Vista client devices** – `C:\Users\username\AppData\Roaming\Sun\SSGD\profile.xml`

Note – This location also contains the user's `hostsvisited` and `certstore.pem` files.

The following settings from the SGD client profile are stored in the location of the user's roaming profile:

| Setting | Profile Entry |
|--------------------------------|---------------|
| Login URL | <url> |
| Add Applications to Start Menu | <mode> |

| Setting | Profile Entry |
|-------------------------|--|
| Automatic Client Login | <autologin><AT> |
| Connect on System Login | <autostart> |
| Connection Failure | <reconnect_mode> <reconnect_attempts> <reconnect_interval> |

The settings that are stored with the user's roaming profile are controlled by the properties file
`/opt/tarantella/var/serverconfig/local/roamingattributes.properties`.

Roaming user profiles are not enabled by default. See [“How to Enable Automatic Installation for Roaming User Profiles” on page 304](#) for details of how to configure SGD to use roaming profiles.

Integrated Mode

This section describes how you can access SGD from the desktop Start or Launch Menu on the client device. Operating SGD in this way is called *Integrated mode*.

When users first connect to an SGD server, they usually start a browser and go to the `http://server.example.com/sgd` URL, where *server.example.com* is the name of an SGD server. They can then log in to SGD and display a webtop. However, once users have logged in, the SGD Client can be configured to use Integrated mode.

When the SGD Client operates in Integrated mode, the links for starting applications are displayed in the desktop Start or Launch Menu, instead of on the webtop. This means that users can run remote applications in the same way as local applications. Depending on how you configure Start Menu integration, there might be no need to use a browser.

Use Integrated mode if your organization prefers not to use Java technology on the client device. See also [“Using SGD Without Java Technology” on page 309](#).

The desktop systems that supported for Integrated mode are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

This section includes the following topics:

- [“Working in Integrated Mode” on page 321](#)
- [“Setting Up the SGD Client for Integrated Mode” on page 322](#)

Working in Integrated Mode

When the SGD Client is in Integrated mode, the user logs in to SGD by clicking the Login link on their desktop Start or Launch Menu.

FIGURE 6-1 Logging In From the Desktop Start Menu



If the user has logged in to more than one SGD server, there is a Login link for each server in the Start or Launch Menu.

Note – To use Integrated mode, you must log in using the Start or Launch Menu. Integrated mode is not available if you start a browser and log in.

Once the user has logged in to SGD, the Start or Launch Menu is updated with the links for the applications they can run through SGD.

FIGURE 6-2 Application Links in the Desktop Start Menu



To start an application, the user clicks the application’s link on the Start or Launch Menu. To start another instance of the application, the user clicks the link again.

Working in Integrated mode simplifies session management. Unlike the webtop, there are no controls for suspending and resuming applications. Instead, when the user logs out, the SGD Client automatically suspends or ends all running application sessions. When the user logs in again, the SGD Client automatically resumes all suspended sessions.

In Integrated mode, users *cannot* start applications with a different user name and password, by pressing and holding down the Shift key when clicking the application's link. See [“Users Can Start Applications With Different User Names and Passwords”](#) on page 212.

Printing is simplified too. Printing is always “on” and print jobs go straight to the printer the user has selected. Unlike the webtop, print jobs cannot be managed individually.

If the user needs to display a webtop, for example to be able to edit their profile, resume a suspended application, or manage printing, they can click the Webtop link on the Start Menu. The user is not prompted to log in, as they already have a user session. The webtop is displayed in their default browser.

If the user has arranged any of their webtop content to display in groups, those groups are also used in the Start or Launch Menu. If the group is configured to hide webtop content, the content does not display in the Start or Launch Menu.

To log out of SGD, the user clicks the Logout link on the Start or Launch Menu.

Setting Up the SGD Client for Integrated Mode

Setting up Integrated mode for the SGD Client involves the following configuration steps:

1. Enable at least one other authentication mechanism.

The user must log in and be authenticated by another authentication mechanism so that SGD can store a user identity and user profile when the user generates an authentication token.

You can use third-party authentication, or any of the other system authentication mechanisms, apart from anonymous user authentication.

See [“Secure Global Desktop Authentication”](#) on page 47.

2. Configure SGD for authentication token authentication.

In Integrated mode, if you configure the SGD Client to log in users in automatically to SGD, an authentication token is used to authenticate the user.

See [“Authentication Token Authentication”](#) on page 323.

3. Enable client profile editing.

Client profile editing must be enabled, to allow users to generate authentication tokens. You can enable profile editing for all users, or just for users that require authentication tokens.

See [“How to Configure Client Profile Editing for Users”](#) on page 313.

4. Configure the client profile for Integrated mode.

Integrated mode must be enabled in the client profile. Other settings in the client profile also affect how Integrated mode works.

See [“Configuring the Client Profile for Integrated Mode” on page 328](#).

5. Applications might have to be configured to give users the best experience.

See [“Configuring Applications for Integrated Mode” on page 329](#).

Authentication Token Authentication

Authentication token authentication enables users to log in to SGD if the SGD Client submits a valid authentication token.

Authentication token authentication can only be used when the SGD Client is operating in Integrated mode and a user has an authentication token.

Authentication token authentication is disabled by default.

This section includes the following topics:

- [“How Authentication Token Authentication Works” on page 323](#)
- [“Authentication Tokens and Security” on page 324](#)
- [“How to Enable Authentication Token Authentication” on page 324](#)
- [“Administering Authentication Tokens” on page 325](#)
- [“Troubleshooting Automatic Logins” on page 327](#)

How Authentication Token Authentication Works

When the SGD Client starts, it submits an authentication token to SGD. The user does not enter a user name or password.

If the authentication token is invalid or the SGD Client does not submit a token, the user is not logged in. The SGD login screen is displayed in a browser, so that the user can log in using another system authentication mechanism.

If the SGD Client submits a valid authentication token, the user is logged in.

User Identity and User Profile

The SGD server stores the authentication token against the identity of the user when they generated their authentication token. This means the user identity and user profile used are those of the authentication mechanism that originally authenticated the user. See [Chapter 2](#) for details of the SGD authentication mechanisms.

Authentication Tokens and Security

When a user generates an authentication token and saves their client profile, the SGD server sends the authentication token to the SGD Client. The SGD Client stores the token in the profile cache on the client device. See [“About the Profile Cache” on page 317](#).

To ensure an authentication token cannot be intercepted and used by a third party, use secure HTTP over Secure Sockets Layer (HTTPS) web servers and enable SGD security services.

When a user generates an authentication token, SGD maintains a record of the tokens issued in a *token cache*. SGD stores the authentication token using the current identity of the user when the token was generated.

When a user logs in with an authentication token, the authentication token enables SGD to “remember” the user’s original identity and user profile. All user sessions and application sessions are managed using the original user identity and user profile.

If the original login becomes invalid, for example because the UNIX system account is disabled or the password has expired, the user can still log in automatically if they have a valid token. However, they cannot run any applications using the invalid credentials.

▼ How to Enable Authentication Token Authentication

1. **In the Administration Console, display the Secure Global Desktop Authentication Configuration Wizard.**

On the Global Settings → Secure Global Desktop Authentication tab, click the Change Secure Global Desktop Authentication button.

2. **On the Third-Party/System Authentication step, ensure the System Authentication check box is selected.**
3. **On the System Authentication - Repositories step, select the Authentication Token check box.**

4. On the **Review Selections** step, check your authentication configuration and click **Finish**.

The Secure Global Desktop Authentication Configuration Wizard closes.

5. On the **Secure Global Desktop Authentication** tab, select the **Token Generation** check box.
6. Click **Save**.

Administering Authentication Tokens

SGD Administrators can use the Administration Console or the `tarantella tokencache` command to administer authentication tokens. The following administration tasks can be done:

- Viewing the tokens in the token cache
- Deleting tokens from the token cache
- Preventing users from generating new tokens

If token generation is enabled, users can generate a new authentication token from the webtop.

▼ How to View Authentication Tokens

You can view the entries in the token cache that belong to a particular user identity or user profile.

- **Use the Administration Console to view the authentication tokens for a user.**
 - On the **Caches → Tokens** tab, use the search to find a user identity if needed.
 - On the **Sessions** tab, click a user identity and then click the **Token** tab.
 - On the **User Profiles** tab, select a user profile and then click the **Tokens** tab.

Tip – On the command line, you can use the `tarantella tokencache list` command to show all entries in the token cache.

▼ How to Delete Authentication Tokens

Deleting a token from the token cache makes the token stored on a client device invalid. If the SGD Client presents an invalid token, the user is prompted to log in with a user name and password. The user must then generate another authentication token if they want to log in automatically.

- **Use the Administration Console to delete authentication tokens.**

- On the Caches → Tokens tab, use the search feature to find a user identity, if needed.
Select the check box next to a token and click Delete.
- On the User Sessions tab, click a user identity and then go to the Token tab.
Click Delete.
- On the User Profiles tab, click a user profile and then go to the Tokens tab.
Select the check box next to a token and click Delete.

Tip – On the command line, you can use the `tarantella tokencache delete` command to delete token cache entries.

Tip – You can run the `tarantella tokencache delete` command on any SGD server in the array. The change is replicated to the other servers in the array.

▼ How to Disable Token Generation

Use this procedure to prevent SGD from issuing new authentication tokens. If authentication token authentication is still enabled, users with existing authentication tokens can still log in to SGD.

1. **In the Administration Console, go to the Global Settings → Secure Global Desktop Authentication tab.**

Deselect the Token Generation check box and click Save.

2. **(Optional) On the command line, use the following command:**

```
$ tarantella config edit --login-autotoken 0
```

▼ How to Generate a New Authentication Token

If a user needs to generate a new authentication token, they must edit their client profile.

1. **Click the Edit button in the Applications area of the webtop and then go to the Client Settings tab.**
2. **Clear the Automatic Client Login box.**
3. **Click Save.**

4. Check the Automatic Client Login box.

5. Click Save.

See [“Setting Up the SGD Client for Integrated Mode”](#) on page 322 for more details about using an authentication token to log in to SGD.

Troubleshooting Automatic Logins

To troubleshoot problems with automatic logins, use the following log filters:

```
server/login/*:autologin%%PID%%.log
server/login/*:autologin%%PID%%.jsl
server/tokencache/*:autologin%%PID%%.log
server/tokencache/*:autologin%%PID%%.jsl
```

The `server/login/*` filter enables you to see when authentication tokens are used for authentication and when they fail.

The `server/tokencache/*` filter enables you to see errors with operations on the token cache. For example, to see why a token is not added to the token cache.

See [“Using Log Filters to Troubleshoot Problems With an SGD Server”](#) on page 391 for more information on configuring and using SGD log filters.

Configuring the Client Profile for Integrated Mode

The following settings in a client profile are applicable when using Integrated mode.

| Setting | Description |
|--------------------------------|--|
| Add Applications to Start Menu | Enables Integrated mode. Causes the SGD Client to add icons to the user's desktop Start or Launch Menu. |
| Automatic Client Login | Enables automatic logins to SGD. If this is disabled, users must log in with a browser. This means they see a webtop <i>and</i> have applications in their desktop Start or Launch Menu. If this is enabled, an authentication token is generated when the client profile is saved. Only users can select this check box. |
| Connect on System Login | If enabled, the SGD Client connects each time the user logs into the desktop system. If Automatic Client Login is also enabled, this gives users a single sign-on experience. |
| Proxy Settings | Proxy server settings can be configured in the client profile itself or obtained from the user's browser. Configuring the settings in the client profile itself reduces the need for a browser. See "Configuring Client Proxy Settings" on page 9 for more details. |

SGD Administrators can configure all these settings apart from the Automatic Client Login.

When configuring Integrated mode, ensure that the login URL in the client profile contains a fully qualified domain name.

All of the available client profile settings for Integrated mode can be configured by both SGD Administrators and users, except for the Automatic Client Login setting.

The Automatic Client Login setting enables automatic logins to SGD, and can only be configured by individual users. This is because when Automatic Client Login is first enabled, SGD generates a unique authentication token for the user when the client profile is saved. The authentication token is stored in the profile cache on the user's client device. This means that users must be able to edit their client profiles, in order to generate an authentication token.

If a user logs in to different SGD servers, they must log in to each SGD server and edit their client profile.

If a user edits their client profile, they must log out of SGD and log in again for the changes to take effect.

To use automatic logins, users click the SGD Login link in their desktop Start menu. If the Connect on System Login check box in the client profile is selected, the SGD Client logs in automatically when a user logs in to their desktop.

Configuring Applications for Integrated Mode

For applications that are configured with a Window Type of Independent Window, closing the window might end or suspend the application session, depending on the setting of the application's Window Close Action attribute.

In Integrated mode, there are no controls for suspending and resuming individual application instances. Applications that are configured to be always resumable are automatically suspended when you log out and resumed when you log in. In the Administration Console, application objects that are always resumable have an Application Resumability setting of General in the Launch tab.

While in Integrated mode, you can only resume a suspended session by displaying a webtop and using the session controls for the application.

You might also want to configure the Number of Instances attribute, to limit the number of instances of applications that users can run.

Webtops

The webtop is a JavaServer Pages™ (JSP™) technology web application. The standard webtop can be customized, or you can develop your own using the SGD web services.

Setting the Language for the Webtop

By default, the SGD web server Welcome Page at `http://server.example.com`, where `server.example.com` is the name of an SGD server, is displayed in English.

To change the default language of the SGD web server Welcome Page, amend the symbolic link `/opt/tarantella/webserver/apache/apache-version/htdocs/index.html`, so that it links to another index page in this directory. For example, to make the default Welcome Page display in Japanese, link to the `index_ja.html` page.

When users log in using a browser at the `http://server.example.com/sgd` URL, where *server.example.com* is the name of an SGD server, the default language used for messages displayed by the login dialog and the webtop is controlled by the `defaultlanguage` parameter setting in the following file:

`/opt/tarantella/webserver/tomcat/tomcat-version/webapps/sgd/WEB-INF/web.xml`

To change the default language, edit this file and replace the parameter value `en` with the language identifier for one of the following supported languages:

| Language | Identifier |
|---------------------|------------|
| English | en |
| French | fr |
| German | de |
| Japanese | ja |
| Korean | ko |
| Simplified Chinese | zh_CN |
| Traditional Chinese | zh_TW |

Save changes to the `web.xml` file and restart the SGD web server.

The default language is also controlled by the Preferred Language in the user's client profile. Whenever the SGD Client is started from the command line, for example when the SGD Client is in Integrated mode, the language specified in the profile is used for messages displayed by the SGD Client, the login dialog, and the webtop. SGD Administrators can set the default language by editing the profiles in their organizational hierarchy. See also [“Client Profile Settings” on page 315](#).

Note – To be able to display text for a locale, users must also have appropriate fonts installed on their client device.

Overriding the Default Language for the Webtop

Individual users can override the default language for the webtop in the following ways:

- On the SGD web server Welcome Page, select a preferred language from the list and then click Log in to access a webtop in that language.

The SGD web server Welcome Page is at `http://server.example.com`, where *server.example.com* is the name of an SGD server.

- Specify a different preferred language in the client profile.
- Log in to SGD using a URL that specifies the preferred language. The URL is `http://server.example.com/sgd/index.jsp?langSelected=lang`, where *lang* is a supported language identifier for SGD and *server.example.com* is the name of an SGD server. Users can manually type this URL in their browser.
- Run the SGD Client from the command line and use the `-preferredlanguage lang` command line argument to set the language, where *lang* is a supported language identifier for SGD. This argument can be used in shortcuts and shell scripts.

Note – When you override the default language, the login URL specified in the user's client profile does not need to be changed. This is usually `http://server.example.com/sgd`, where *server.example.com* is the name of an SGD server.

SGD Servers, Arrays, and Load Balancing

This chapter describes how to configure and monitor Oracle Secure Global Desktop (SGD) servers and arrays. Some system administration features of SGD, such as the Administration Console, log filters, and installation backups are also covered.

This chapter includes the following topics:

- [“Arrays” on page 333](#)
- [“Load Balancing” on page 352](#)
- [“SGD Web Server and Administration Console” on page 379](#)
- [“Monitoring and Logging” on page 386](#)
- [“SGD Server Certificate Stores” on page 408](#)
- [“SGD Installations” on page 411](#)
- [“Troubleshooting Arrays and Load Balancing” on page 419](#)

Arrays

In SGD, an *array* is a collection of SGD servers that share configuration information.

Arrays have the following benefits:

- Users and application sessions are *load-balanced* across the array. To scale to more users, simply add more SGD servers to the array. See [“Load Balancing” on page 352](#) for more details.
- With more than one server, there is no single point of failure. You can decommission a server temporarily with the minimum of disruption to your users.

- Configuration information, including all the objects in your organizational hierarchy, is replicated to all array members. All array members have access to all information.

Users see the same webtop and can resume applications, no matter which SGD server they log in to.

This section includes the following topics:

- [“The Structure of an Array” on page 334](#)
- [“Replicating Data Across the Array” on page 334](#)
- [“Communication Between Array Members” on page 335](#)
- [“Secure Intra-Array Communication” on page 336](#)
- [“Managing Arrays and SGD Servers” on page 338](#)
- [“Array Resilience” on page 338](#)
- [“Configuring Arrays” on page 343](#)
- [“Configuring Array Resilience” on page 348](#)

The Structure of an Array

An array contains the following:

- **One primary server.** This server is the authoritative source for global SGD information, and maintains the definitive copy of the organizational hierarchy, called the local repository.
- **One or more secondary servers.** The primary server replicates information to these servers.

A single, *standalone* server is considered to be the primary server in an array with no secondary servers.

SGD servers in an array might run different operating systems. However, all the array members must run the same version of SGD.

As the SGD servers in an array share information about user sessions and application sessions, it is important to synchronize the clocks on the SGD hosts. Use Network Time Protocol (NTP) software, or the `rdate` command, to ensure the clocks on all SGD hosts are synchronized.

Replicating Data Across the Array

When the primary server replicates data to the secondary servers, it replicates the following data:

- The local repository
- Session information
- Configuration information, including global configuration
- Client profiles created by SGD Administrators
- User preferences created by SGD users from the webtop
- The application server password cache
- The token cache
- Resource files, such as application server login scripts

Apart from the resource files, any changes to the above data are replicated immediately.

The synchronization of resource files occurs once daily, and only while the servers are running. The resource files that are synchronized are the files from the `/opt/tarantella/var/serverresources` directory. Only add, modify, or delete the files in these directories on the primary server.

The time and effort that it takes to synchronize an array is directly proportional to the size of the array.

Resource synchronization can be scheduled to take place at a time of your choice. In the Administration Console, this is configured with the Daily Resource Synchronization Time attribute on the Performance tab for each SGD server.

Communication Between Array Members

In the array, each SGD server has a peer Domain Name System (DNS) name and one or more external DNS names. SGD servers always use peer DNS names to communicate with each other. You also use peer DNS names when specifying array members in the SGD configuration tools. External DNS names are only used by SGD Clients when connecting to SGD servers. See [“DNS Names” on page 3](#) more details.

Connections between the SGD servers in an array are made on Transmission Control Protocol (TCP) port 5427. Unless explicitly enabled, this connection is not encrypted. The connection between SGD servers in an array can be encrypted by using secure intra-array communication. See [“Secure Intra-Array Communication” on page 336](#).

Each server in the array has a record of the peer DNS names of all the SGD servers in an array. A server only accepts connections on TCP port 5427 if the following occurs:

- The connection is from an array member, according to its own records.

- A shared secret, known only to array members, is used to authenticate connections between array members. Secret Key Identification (SKID) authentication is used. SKID authentication does not encrypt the data.

Most connections are made from the primary server to a secondary server. These connections replicate data to keep the array synchronized. However, array members must be able to communicate directly with other array members.

Secure Intra-Array Communication

In a standard installation, the data transmitted between the SGD servers in an array is not encrypted. SGD Administrators can secure the connections between array members using SSL. Using SSL for these connections ensures the integrity of the data as follows:

- Communication only takes place between SGD servers that have authenticated to each other
- Data is encrypted before transmission
- Data can be checked to ensure that it has not changed during transmission

Using SSL in this way is known as *secure intra-array communication*.

Secure intra-array communication can only be enabled on an SGD server that is not joined with other SGD servers in an array. When secure intra-array communication is enabled for an array, an SGD server can only join the array if it also has secure intra-array communication enabled. See [“How to Enable Secure Intra-Array Communication” on page 344](#) for details.

Using secure intra-array communication means that each SGD server in the array has to have a valid SSL certificate that has been signed by a trusted certificate authority (CA).

As the SSL certificates used for secure intra-array communication are used only internally by SGD, the primary SGD server in the array acts as the CA. The primary SGD server has a self-signed CA certificate and a private key. All secondary SGD servers in the array have a copy of the primary SGD server’s CA certificate in a trusted certificate store, the truststore.

All SGD servers in the array, including the primary, have an SSL certificate and a private key. The SSL certificate is signed with the primary SGD server’s CA certificate and contains a common name (CN) which is the peer DNS name of the SGD server. As the SSL certificates are created using a self-signed CA certificate, they cannot be used to secure any other SGD-related connection. These certificates are referred to as *peer SSL certificates* to distinguish them from other types of SSL certificates.

When one SGD server in the array connects to another, including when using an administration tool, the SGD server being connected to presents its peer SSL certificate as part of the SSL negotiation. The connecting server evaluates the peer SSL certificate and checks the following:

- The CN of the certificate matches the peer DNS name of the connecting server
- The expiry date of the certificate
- The issuer of the certificate, which must be the CA certificate of the primary

If the peer SSL certificate is valid, a secure connection is established.

When you enable secure intra-array communication, SGD automatically generates and distributes the CA and peer SSL certificates to the members of the array. Whenever there is a change in the array structure, SGD automatically updates the CA and peer SSL certificates. The following table summarizes what happens.

| Array Change | Action |
|------------------------------|--|
| Server joins the array | <ol style="list-style-type: none"> 1. The primary SGD server CA certificate is installed on the new secondary server. 2. The new secondary SGD server obtains a new peer SSL certificate signed with the primary SGD server CA certificate. |
| Server leaves the array | <ol style="list-style-type: none"> 1. The detached SGD server becomes the primary SGD server in an array containing one server. 2. The detached SGD server creates a new CA certificate for itself. 3. The detached SGD server creates a new peer SSL certificate for itself. |
| New primary server appointed | <ol style="list-style-type: none"> 1. The new primary SGD server generates a new CA certificate. 2. The new primary CA certificate is installed on all secondary SGD servers. 3. All SGD servers obtain a new peer SSL certificate signed with the new primary SGD server CA certificate. |

SGD Administrators can use the `tarantella security peerca --show` command to view certificates in the truststore. The truststore contains the primary SGD server's CA certificate.

By default, SGD uses the `TLS_RSA_WITH_AES_128_CBC_SHA` cipher suite for secure intra-array communication. For details on how to change the cipher suite, see [“How to Change the Cipher Suite for Secure Intra-Array Communication” on page 347](#).

Managing Arrays and SGD Servers

You add and remove SGD servers from an array using Secure Global Desktop Servers tab in the Administration Console or by using the `tarantella array` command. It is best to perform all array operations on the primary SGD server in the array. For details on configuring arrays, see the following:

- [“How to Add a Server to an Array \(Secure Intra-Array Communication Enabled\)” on page 344](#)
- [“How to Add a Server to an Array \(Secure Intra-Array Communication Disabled\)” on page 345](#)
- [“How to Change the Primary Server in an Array” on page 346](#)
- [“How to Remove a Server From an Array” on page 347](#)

In the Administration Console, the attributes on the Global Settings tabs are the settings that apply to the array as a whole, for example how users authenticate to SGD. [Appendix A](#) has details of all the global settings. If you click the name of an SGD server on the Secure Global Desktop Servers tab, you display the attributes that apply only to that SGD server, for example the server’s external DNS names. [Appendix B](#) has details of all the server-specific settings.

On the command line, you list and edit global settings or server-specific settings using the `tarantella config` command.

Array Resilience

Array resilience is a feature where the loss of the primary server in an SGD array is handled automatically. The primary server can become unavailable either because of network problems, or because the SGD server goes down.

This section includes the following topics:

- [“How Array Resilience Works” on page 338](#)
- [“Examples of How Array Resilience Works” on page 341](#)

How Array Resilience Works

Array resilience consists of the following stages:

- **Failover stage.** When the primary server becomes unavailable, the array is reconfigured automatically into one or more arrays, each with their own primary server. See [“Failover Stage” on page 339](#).

- **Recovery stage.** When the original primary server becomes available, the original array formation is recreated. This can be done automatically or manually. See [“Recovery Stage” on page 340](#).

Failover Stage

If array failover is enabled for an array, the failover stage starts automatically after the primary server has been unavailable for a user-configurable time period, called the *grace period*. The default grace period is 10 minutes.

The grace period is calculated from the values for the Monitor Interval (`--array-monitortime`) and Monitor Attempts (`--array-maxmonitors`) attributes, as follows:

Grace period = Monitor Interval x Monitor Attempts

Using the default settings:

Grace period = 60 seconds x 10 = 600 seconds, or 10 minutes

The failover stage uses the *backup primaries list* to select a secondary server to promote to be the new primary server in the array. The backup primaries list is a list of secondary servers for the array, with the priority decreasing from top to bottom. If available, the highest priority secondary server in this list is contacted and promoted to be the new primary server for the array.

The new primary server must be contacted within a time period called the *find new primary timeout*. If the new primary cannot be contacted during this timeout period, the next server in the backup primaries list is contacted.

The find new primary timeout period is calculated from the values for the Find Primary Interval (`--array-resubmitfindprimarywait`) and Find Primary Attempts (`--array-resubmitfindprimarymax`) attributes, as follows:

Find new primary timeout = Find Primary Interval x Find Primary Attempts

Using the default settings:

Find new primary timeout = 60 seconds x 3 = 180 seconds, or 3 minutes

Only SGD servers that are in the backup primaries list can be selected for promotion to be the new primary server.

When you build an SGD array, a backup primaries list is created for you automatically. If you add a secondary server to the array, an entry is added at the end of the list. If you remove a secondary server from the array, the entry for the server is removed from the list.

The backup primaries list is stored on each SGD server in the array. Any changes made to the list are copied to each SGD server in the array.

If the backup primaries list is empty, all SGD servers in the array become standalone servers after array failover.

When the failover stage has completed, the array is said to be in a *repaired* state.

The `tarantella status` command indicates if an array is in a repaired state. You can use the `--originalstate` option of this command to list the members of the array before it was repaired. See [“Showing Status Information For an SGD Array” on page 420](#) for more details about using `tarantella status` to display status information for an array.



Caution – During the failover stage, do not change the array formation, or any array resilience settings. The original array formation might not be recreated successfully by the recovery stage if you do so.

Recovery Stage

If the original primary server becomes available when the array is in a repaired state, the recovery stage starts automatically.

By default, the recovery stage restores the original primary server as the primary server for the array. You can use the Action When Failover Ends (`--array-primaryreturnaction`) attribute to determine how the array is reconfigured during the recovery stage.

In some situations after using array resilience you might have to rebuild an array manually. This is called *manual recovery*.

For example, if the recovery stage has failed to recreate the original array formation automatically you can rebuild the original array manually, starting from single, standalone SGD servers. You use the `tarantella array clean` command to do this.



Caution – After you run `tarantella array clean` on the primary server in an SGD array, the secondary servers will not be able to contact the primary server.

If an array splits into more than two arrays during the failover stage, the original array formation cannot be recreated automatically. Manual recovery must be used.

Examples of How Array Resilience Works

There are many possible array resilience scenarios, where the primary server becomes unavailable for one or more servers in the SGD array. This section includes examples of how array resilience works in the following scenarios:

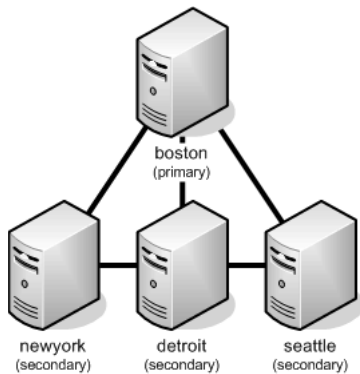
- [“Primary Server Goes Down” on page 341](#)
- [“Array Splits into Two Arrays” on page 342](#)

In the following examples, the domain `example.com` has a four-node array of SGD servers:

- **Primary server** – `boston`
- **Secondary servers** – `newyork`, `detroit`, `seattle`

[FIGURE 7-1](#) shows the original network configuration before using array resilience.

FIGURE 7-1 Original Network Configuration



Primary Server Goes Down

A typical sequence of events for array resilience when the primary server in an SGD array goes down is as follows:

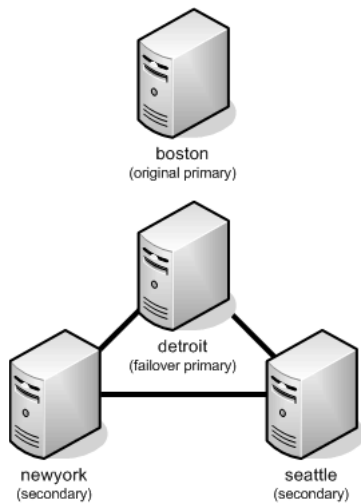
1. The primary server, `boston`, goes down and is unavailable to any of the secondary servers in the array.
2. If `boston` is still unavailable after the grace period has elapsed, the failover stage begins.
3. The first available secondary server from the array's backup primaries list is promoted to be the new primary server for the array.

- Each of the existing secondary servers are reconfigured automatically to work with the new primary server. The array becomes a three-node array. The array is now in a repaired state.

FIGURE 7-2 shows the network configuration after the failover stage.

- When `boston` becomes available again, the recovery stage begins. By default, `boston` automatically rejoins the array as the primary server.
- The other servers in the array are reconfigured automatically to work with the new primary server, `boston`.

FIGURE 7-2 Network Configuration After the Failover Stage When the Primary Server Goes Down



Array Splits into Two Arrays

A typical sequence of events for array resilience when an SGD array splits into two arrays is as follows:

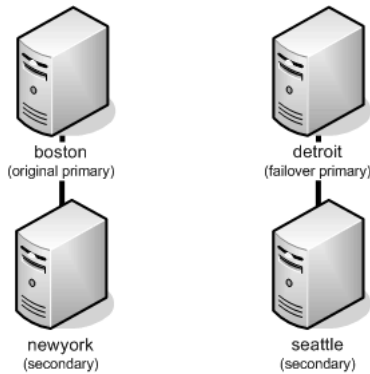
- Because of network problems, the primary server, `boston`, can only contact the `newyork` secondary server. The remaining secondary servers in the array, `seattle` and `detroit`, cannot be contacted.
- After the grace period has elapsed, if the primary server still cannot contact `seattle` and `detroit`, the failover stage begins.

3. The original array remains as a four-node array, but with the `seattle` and `detroit` servers reported as uncontactable in the array. The same primary server, `boston`, is used but the original array now has only a single contactable secondary server, `newyork`.
4. The secondary servers `seattle` and `detroit` can contact each other. These servers join to form a new two-node array. The first available secondary server from the backup primaries list is promoted to be the primary server for this array.

[FIGURE 7-3](#) shows the network configuration after the failover stage.

5. Network problems are fixed. The recovery stage begins. By default, the two arrays join together. The original array formation is recreated automatically, with `boston` as the primary server.

FIGURE 7-3 Network Configuration After the Failover Stage When the Array Splits into Two Arrays



Configuring Arrays

Configuring an array involves the following steps:

1. Add SGD servers to the array.

Before building an array, you might want to enable secure intra-array communication, [“How to Enable Secure Intra-Array Communication” on page 344](#).

How you add servers to an array depends on whether or not you are using secure intra-array communication, see the following:

- [“How to Add a Server to an Array \(Secure Intra-Array Communication Enabled\)” on page 344](#)

- [“How to Add a Server to an Array \(Secure Intra-Array Communication Disabled\)” on page 345](#)
- 2. Change the structure of an array.
See the following:
 - [“How to Change the Primary Server in an Array” on page 346](#)
 - [“How to Remove a Server From an Array” on page 347](#)
- 3. (Optional) Change the cipher suite used for secure intra-array communication.
See [“How to Change the Cipher Suite for Secure Intra-Array Communication” on page 347](#).

▼ How to Enable Secure Intra-Array Communication

Secure intra-array communication can only be enabled on an SGD server that is not joined with other SGD servers in an array.

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

You can only enable secure intra-array communication from the command line.

1. **Log in as superuser (root) on the SGD server.**
2. **Stop the SGD server.**
3. **Enable secure intra-array communication.**

Use the following command:

```
# tarantella config edit \  
--tarantella-config-security-peerssl-enabled 1
```

4. **Start the SGD server.**

▼ How to Add a Server to an Array (Secure Intra-Array Communication Enabled)

When secure intra-array communication is enabled for an array, an SGD server can only join the array if it also has secure intra-array communication enabled.

The clock on the server joining the array must be in synchronization with the clocks on the other servers in the array. If the time difference is more than one minute, the array join operation fails.

1. Log in to the SGD server that you want to add to the array.
2. Display the fingerprint of the SGD server's CA certificate.

Use the following command:

```
$ tarantella security peerca --show
```

3. Make a note of the fingerprint of the SGD server's CA certificate.
4. Log in to the primary SGD server in the array.
5. Join the SGD server to the array as a secondary server.

Use the following command to add the SGD server.

```
$ tarantella array join --secondary serv
```

It is best to use a fully-qualified DNS name for *serv*.

You are prompted to trust the secondary SGD server's CA certificate, and the fingerprint of the certificate is displayed.

6. Check that the fingerprint is correct and complete the array join.

Check that the certificate fingerprint matches the fingerprint displayed in [Step 2](#). This is important as it verifies that the primary SGD server is communicating with the genuine secondary SGD server.

If the fingerprints match, complete the array join by accepting the secondary SGD server's CA certificate.

7. Check the status of the array.

Use the `tarantella status` command to check the status of the array.

▼ How to Add a Server to an Array (Secure Intra-Array Communication Disabled)

The server joining the array must be a standalone server. In other words, the server must be in an array on its own.

Ensure that the clock on the server joining the array is in synchronization with the clocks on the other servers in the array. If the time difference is more than one minute, the add server operation fails.

1. Log in to the Administration Console on the primary SGD server.
2. Go to the Secure Global Desktop Servers tab.

3. In the Secure Global Desktop Server List, click the Add button.

The Add a Secure Global Desktop Server screen is displayed.

Tip – You can also use the `tarantella array join` command to add an SGD server to an array.

4. Enter the DNS name of an SGD server in the DNS Name field.

You must use a fully-qualified DNS name.

5. Enter the user name and password of an SGD Administrator in the User Name and Password fields.

6. Click Add.

The Secure Global Desktop Servers tab is displayed.

The Secure Global Desktop Servers tab shows messages advising you to wait for the server change and synchronization processes to complete.

Note – After making a change to the structure of an array, wait until SGD has copied the changes to all the SGD servers in the array before making any further changes. Run the `tarantella status` command on the primary SGD server to check the status of the array.

If the server you add has been load balancing application servers using Advanced Load Management, you must do a warm restart, `tarantella restart sgd -warm`, of the new server after it has joined the array. See also [“How Advanced Load Management Works” on page 371](#).

▼ How to Change the Primary Server in an Array

1. Log in to the Administration Console on the primary SGD server.

2. Go to the Secure Global Desktop Servers tab.

3. In the Secure Global Desktop Server List, click the Make Primary button.

Tip – You can also use the `tarantella array make_primary` command to change the primary server in an array.

4. When prompted, click OK.

The Secure Global Desktop Servers tab shows messages advising you to wait for the server change and synchronization processes to complete.

The previous primary server becomes a secondary server.

Note – After making a change to the structure of an array, wait until SGD has copied the changes to all the SGD servers in the array before making any further changes. Run the `tarantella status` command on the primary SGD server to check the status of the array.

▼ How to Remove a Server From an Array

To remove the primary server from an array, you must first make another server the primary server and then remove the old primary server.

1. **Log in to the Administration Console on the primary SGD server.**
2. **Go to the Secure Global Desktop Servers tab.**
3. **In the Secure Global Desktop Server List, click the Remove button.**

Tip – You can also use the `tarantella array detach` command to remove an SGD server from an array.

4. When prompted, click OK.

The Secure Global Desktop Servers tab shows messages advising you to wait for the server change and synchronization processes to complete.

Note – After making a change to the structure of an array, wait until SGD has copied the changes to all the SGD servers in the array before making any further changes. Run the `tarantella status` command on the primary SGD server to check the status of the array.

▼ How to Change the Cipher Suite for Secure Intra-Array Communication

Ensure that no users are logged in to the SGD servers in the array and that there are no running application sessions, including suspended application sessions.

1. **Stop all the SGD servers in the array.**

2. Log in as superuser (root) on the primary SGD server in the array.

3. Specify the cipher suite.

Use the following command:

```
# tarantella config edit \  
--tarantella-config-security-peerssl-ciphers cipher-suite
```

where *cipher-suite* is the JSSE name of a cipher suite.

The following table lists the available cipher suites.

| JSSE Name | Cipher Suite |
|-------------------------------|---------------------------|
| TLS_RSA_WITH_AES_256_CBC_SHA | RSA_WITH_AES_256_CBC_SHA |
| TLS_RSA_WITH_AES_128_CBC_SHA | RSA_WITH_AES_128_CBC_SHA |
| SSL_RSA_WITH_3DES_EDE_CBC_SHA | RSA_WITH_3DES_EDE_CBC_SHA |
| SSL_RSA_WITH_RC4_128_SHA | RSA_WITH_RC4_128_SHA |
| SSL_RSA_WITH_RC4_128_MD5 | RSA_WITH_RC4_128_MD5 |
| SSL_RSA_WITH_DES_CBC_SHA | RSA_WITH_DES_CBC_SHA |

The default setting is TLS_RSA_WITH_AES_128_CBC_SHA.

4. Start all the SGD servers in the array.

Configuring Array Resilience

Configuring array resilience involves the following steps:

1. Enable array failover for the array.

Array failover is disabled by default for an SGD array.

■ See [“How to Enable Array Failover for an Array” on page 349](#).

2. (Optional) Configure the array failover grace period.

Array failover starts automatically after the grace period has elapsed.

■ See [“How to Configure the Array Failover Grace Period” on page 349](#).

3. (Optional) Configure the backup primaries list.

The backup primaries list determines which secondary server is promoted to be the new primary server.

For more details, see the following:

- [“How to Show the Backup Primaries List for an Array” on page 350](#)
 - [“How to Add an Entry to the Backup Primaries List” on page 350](#)
 - [“How to Change the Position of an Entry in the Backup Primaries List” on page 350](#)
 - [“How to Delete an Entry From the Backup Primaries List” on page 351](#)
4. (Optional) Configure the find new primary timeout period.

If the new primary cannot be contacted during this timeout period, the next server in the backup primaries list is contacted.

 - See [“How to Configure the Find New Primary Timeout” on page 351](#).
 5. (Optional) Configure the recovery stage.

By default, the recovery stage recreates the original array formation automatically when the original primary server becomes available.

 - See [“How to Configure the Action When Failover Ends” on page 351](#).

You can use manual recovery to recreate the original array formation manually.

 - See [“How to Rebuild an Array Manually” on page 352](#).

▼ How to Enable Array Failover for an Array

1. In the Administration Console, go to the Global Settings → Resilience tab.
2. Enable array failover for the SGD array.

Select the Array Failover check box.

Tip – You can also use the `tarantella config edit` command to enable the Array Failover (`--array-failoverenabled`) attribute.

▼ How to Configure the Array Failover Grace Period

1. In the Administration Console, go to the Global Settings → Resilience tab.
2. Configure the grace period.

Type values for the Monitor Interval and Monitor Attempts attributes.

For example, to change the grace period to 120 seconds, or 2 minutes, set the Monitor Interval attribute to 60 and the Monitor Attempts attribute to 2.

The default grace period is 10 minutes.

Tip – You can also use the `tarantella config edit` command to configure the Monitor Interval (`--array-monitortime`) and Monitor Attempts (`--array-maxmonitors`) attributes.

▼ How to Show the Backup Primaries List for an Array

1. In the Administration Console, go to the Global Settings → Resilience tab.
2. View the entries in the backup primaries list.

The Backup Primaries table shows the backup primaries list for the array.

Tip – You can also use the `tarantella array list_backup primaries` command to show the backup primaries list for an array.

▼ How to Add an Entry to the Backup Primaries List

1. In the Administration Console, go to the Global Settings → Resilience tab.
2. Add an entry to the backup primaries list.

- a. Click the New button in the Backup Primaries table.

The Available Secondaries table is displayed, showing the available secondary servers that are not in the backup primaries list.

- b. Select a server in the Available Secondaries table and click Add.

The Backup Primaries table is updated automatically.

Tip – You can also use the `tarantella array add_backup_primary` command to add an entry to the backup primaries list.

▼ How to Change the Position of an Entry in the Backup Primaries List

1. In the Administration Console, go to the Global Settings → Resilience tab.
2. Change the position of an entry in the backup primaries list.

Select the server in the Backup Primaries table and click Move Up or Move Down.

Tip – You can also use the `tarantella array edit_backup_primary` command to change the position of an entry in the backup primaries list.

▼ How to Delete an Entry From the Backup Primaries List

1. **In the Administration Console, go to the Global Settings → Resilience tab.**
2. **Delete an entry in the backup primaries list.**
Select the server in the Backup Primaries table and click Delete.

Tip – You can also use the `tarantella array remove_backup_primary` command to delete an entry from the backup primaries list.

▼ How to Configure the Find New Primary Timeout

1. **In the Administration Console, go to the Global Settings → Resilience tab.**
2. **Configure the find new primary timeout period.**
Type values for the Find Primary Interval and Find Primary Attempts attributes.
For example, to change the find new primary timeout to 60 seconds, or 1 minute, set the Find Primary Interval attribute to 60 and the Find Primary Attempts attribute to 1.
The default find new primary timeout period is 3 minutes.

Tip – You can also use the `tarantella config edit` command to configure the Find Primary Interval (`--array-resubmitfindprimarywait`) and Find Primary Attempts (`--array-resubmitfindprimarymax`) attributes.

▼ How to Configure the Action When Failover Ends

1. **In the Administration Console, go to the Global Settings → Resilience tab.**
2. **Configure how the array is reconfigured when the original primary server becomes available.**
Select the required option for the Action When Failover Ends attribute.
 - To accept the original primary server back into the array as the primary server, select the Restore original primary option.

The original primary server, and any attached secondary servers, rejoin the array. The original primary server is restored as the primary server for the array. The current primary server becomes a secondary server. This is the default setting.

- To exclude the original primary server from the array, select the Do not restore original array option.

The original primary server, and any attached secondary servers, do not rejoin the array. The original primary server, and any attached secondary servers, stay in the state they were in after the failover stage.

- To accept the original primary server back into the array as a secondary server, select the Restore array with a new primary option.

The original primary server, and any attached secondary servers, rejoin the array as secondary servers.

Tip – You can also use the `tarantella config edit` command to configure the Action When Failover Ends (`--array-primaryreturnaction`) attribute.

▼ How to Rebuild an Array Manually

1. Remove all array state information.

Run the following command on each SGD server in the array.

```
$ tarantella array clean
```

By default, the `tarantella array clean` command deletes all array information and configures the SGD server as a single, standalone server. Use the `--contactmembers` option of this command if you want the SGD server to remain in an array with other SGD servers that are contactable and that report the same array membership.

2. Rebuild the array manually.

Use the `tarantella array` command. See [“Managing Arrays and SGD Servers” on page 338](#) for details of how to do this.

Load Balancing

Load balancing helps you scale up to support more users so that they receive a reliable and high-performance service without any single point of failure.

SGD supports the following load balancing mechanisms:

- **User session load balancing** – Determines which SGD server in the array a user logs in to
See “[User Session Load Balancing](#)” on page 353 for details.
- **Application session load balancing** – Determines which SGD server in the array manages an application session for a user
See “[Application Session Load Balancing](#)” on page 361 for details.
- **Application load balancing** – Determines which application server runs an application for a user
See “[Application Load Balancing](#)” on page 362 for details.
- **Load balancing groups** – Tries to deliver the best possible user experience by choosing SGD servers and application servers linked by a fast network
See “[Load Balancing Groups](#)” on page 363 for details.

User Session Load Balancing

User session load balancing is concerned with choosing a SGD server to log in to. Users can log in to any SGD server in an array and access the same applications.

User session load balancing happens before the first connection is made to SGD. You can use a number of mechanisms to choose an appropriate SGD server, for example:

- The SGD Gateway
- Round-robin or Dynamic DNS
- An external hardware load balancer
- The SGD load-balancing JavaServer Pages™ (JSP™) technology page
- Allocate different SGD servers to different departments and give one Uniform Resource Locator (URL) to each department

When load balancing user sessions, the most important factor is *session persistence*. A user session begins when a user logs in to an SGD server and the session is owned by that server. As the user interacts with SGD, further requests are sent over the Hypertext Transfer Protocol (HTTP) connection to the SGD server. If network connections are load-balanced, HTTP requests might be directed to any SGD server in the array. If an HTTP request goes to an SGD server that does not own the user session, the following can occur:

- The user session is transferred to that SGD server and the windows of any running applications might disappear. This is sometimes called *session grabbing*.
- The visible state of the user’s session is displayed incorrectly.

To load balance user sessions successfully, HTTP requests must *persist* so that they always go to the correct SGD server.

In a default SGD installation, additional configuration is required to make HTTP connections persistent. SGD supports the following mechanisms:

- **The SGD Gateway.** The SGD Gateway contains an Apache web server that manages the load balancing of HTTP connections to SGD.
- **The load-balancing JSP technology page.** The load-balancing JSP technology page contains a JavaScript™ technology script that sets a cookie, and that cookie is used to redirect HTTP requests to the correct server.

The Gateway provides the best solution for load balancing user sessions. The load-balancing JSP technology page might not be available in future releases of SGD.

Instructions on how to install, configure, and use the SGD Gateway are included in the *Oracle Secure Global Desktop 4.6 Gateway Administration Guide*.

The load-balancing JSP technology page can only be used if the following conditions are met:

- Browsers must have cookies enabled and JavaScript technology enabled
- The SGD Client must not be in Integrated mode

The load-balancing JSP technology page can be used in the following ways:

- The JSP technology page selects an SGD server from a list using a round-robin mechanism.

To use the JSP technology page in this way, you configure the load-balancing JSP technology page on *one member of the array*. See [“Using The Load-Balancing JSP Technology Page to Distribute User Sessions” on page 354](#).

- The JSP technology page supports an external mechanism for selecting an SGD server.

To use the JSP technology page in this way, you configure the load-balancing JSP technology page on *every member of the array*. See [“Using an External Mechanism to Distribute User Sessions” on page 356](#).

Using The Load-Balancing JSP Technology Page to Distribute User Sessions

To use the load-balancing JSP technology page to distribute user sessions, *one member of the array* acts as the load distribution server. Typically this is the primary SGD server in the array. On the load distribution server, you configure the load-balancing JSP technology page with a list of SGD servers that can host user sessions. Users connect to the load distribution server and the load-balancing JSP technology page redirects them to an SGD server in the list using a round-robin mechanism.

Users must connect to the load distribution server using a URL that connects to the load-balancing JSP technology page, usually this is `http://server.example.com/sgd`, where *server.example.com* is the name of an SGD server.

To use Hypertext Transfer Protocol over Secure Socket Layer (HTTPS) connections, you configure SGD as follows:

- Use an HTTP connection to the load distribution server.
- Configure HTTPS URLs for the SGD servers in the load-balancing JSP technology page.

▼ How to Configure the Load-Balancing JSP Technology Page to Distribute User Sessions

Select *one member of the array* to act as the load distribution server. The following procedure uses the primary SGD server in the array.

1. Log in as superuser (root) on the *primary* SGD server in the array.
2. Copy the load-balancing JSP technology pages to the `/sgd` web application directory.

For example:

```
# cd /opt/tarantella/webserver/tomcat/tomcat-version/webapps/sgd/  
# cp -rp admin/loaddist/ swcd/
```

Note – When you copy the files, use the `-p` option to preserve the file permissions.

3. Edit the load-balancing JSP technology page.

The load-balancing JSP technology page is `swcd.jsp`.

- a. Add the external DNS names of the SGD servers to be load balanced.

See “[Configuring External DNS Names](#)” on page 5 for details.

Amend the `hosts = new Array` section, for example:

```
hosts[0] = "http://www1.example.com"  
hosts[1] = "http://www2.example.com"  
...  
hosts[4] = "http://www5.example.com"
```

If you are using secure connections, ensure the URLs begin with `https://`.

Only include the primary SGD server in the list if you want the primary server to host user sessions.

b. Set the LBHOST variable.

Remove the first comment marks (//) as follows:

```
var LBHOST = null // Not in Load Balancer/Round Robin DNS mode
```

c. Save the changes.

4. Configure the SGD entry point JSP technology page to use the load-balancing JSP technology page.

The entry point JSP technology page is `index.jsp`.

a. Change the first line to the following:

```
<%@ include file="swcd/swcd.jsp" %>
```

b. Save the change.

Using an External Mechanism to Distribute User Sessions

When using an external mechanism, such as a hardware load balancer or round-robin DNS, for load balancing user sessions, the following factors are important:

- **External DNS names.** The SGD servers in the array must be directly accessible using their external DNS names. If an external load balancer is acting as a firewall, a switch, or a router, it must be configured to allow access using the external DNS names. See [“Configuring External DNS Names” on page 5](#).
- **Adaptive Internet Protocol (AIP) connections.** AIP connections must go to the SGD server that is hosting the application session. An external load balancer must not distribute the connections to different SGD servers in the array.
- **AIP is not HTTP.** If you enable SGD security services, AIP connections are encrypted using Secure Sockets Layer (SSL). If an external load balancer decrypts the SSL for an AIP request, it cannot handle the remaining contents.
- **URL rewriting.** An external load balancer can be configured to rewrite URLs. The impact of a connection to SGD using a URL containing a host name that does not match the external DNS name of the SGD server is undefined.
- **Virtual hosting multiple HTTPS connections.** To use HTTPS connections to an external load balancer and the SGD web server requires two SSL certificates: one for the load balancer DNS name and one for the external DNS name of the SGD server. You can only do this by using virtual hosting to create two web servers on the same host, each with a different DNS name. However, the web servers must use either different TCP ports or different Internet Protocol (IP) addresses.

To use an external mechanism to distribute user sessions, you configure the load-balancing JSP technology page on *every SGD server in the array*.

If you are using a hardware load balancer, the load balancer must be configured to allow access to the SGD servers using their external DNS names. Typically the load balancer is also an SSL accelerator. With this configuration, the connection to SGD is processed as follows:

1. Users make HTTPS connections to the load balancer DNS name.
2. The load balancer decrypts the SSL request and forwards it as an HTTP request to the external DNS name of the selected SGD server.
3. The load-balancing JSP technology page on the SGD server checks for the load-balancing cookie and redirects the HTTP request to the correct SGD server as needed.

Users must connect to SGD using a URL that contains the DNS name of the load balancer, for example `https://loadbalancer.indigo-insurance.com/sgd`.

If SGD security services are enabled, and the external load balancer is configured to decrypt SSL connections and forward them as unencrypted connections, you must configure each SGD server in the array to accept plain text connections on the secure port. See [“Using External SSL Accelerators” on page 40](#) for details.

To use HTTPS connections between the load balancer and the SGD servers, ensure that the URLs in the load-balancing JSP technology page begin with `https://`. Then perform *either* of the following configurations:

- Configure the external load balancer to terminate the load-balanced HTTPS connection and then regenerate the connection as an HTTPS connection to the external DNS name of the SGD server.
- Assign an additional IP address to the SGD host and configure the host to use this address. Configure the SGD web server to listen on the additional IP address and associate the SSL certificate for the load balancer with this address. Configure the SGD web server to associate the external DNS name of the SGD server with the original IP address of the server. Configure the load balancer to use the additional IP addresses.

Using SGD in firewall traversal mode can also help to simplify the configuration needed when using an external load balancer. With firewall traversal, the HTTP and AIP connections to SGD are made over a single port, usually TCP port 443. See [“Firewall Traversal” on page 24](#).

▼ How to Configure the Load-Balancing JSP Technology Page for an External Load Balancing Mechanism

The following procedure is an example of how to configure the load-balancing JSP technology page on an SGD server when using an external load balancing mechanism.

All the SGD servers in the array must be configured in the same way.

1. **Log in as superuser (root) on the SGD host.**
2. **Copy the load-balancing JSP technology page files to the /sgd web application directory.**

For example:

```
# cd /opt/tarantella/webserver/tomcat/tomcat-version/webapps/sgd/
# cp -rp admin/loaddist/ swcd/
```

Note – When you copy the files, use the `-p` option to preserve the file permissions.

3. **Edit the load-balancing JSP technology page.**

The load-balancing JSP technology page is `swcd.jsp`.

- a. **Add the external DNS names of the SGD servers to be load balanced.**

See [“Configuring External DNS Names” on page 5](#) for details.

Amend the `hosts = new Array` section, for example:

```
hosts[0] = "http://www1.example.com"
hosts[1] = "http://www2.example.com"
...
hosts[4] = "http://www5.example.com"
```

- b. **Set the LBHOST variable.**

Remove the first comment marks (`//`) and enter the external DNS name of the SGD server, for example:

```
var LBHOST = "http://www1.example.com" // LB mode
```

- c. **Save the changes.**
4. **Configure the SGD entry point JSP technology page to use the load-balancing JSP technology page.**

The entry point JSP technology page is `index.jsp`.

- a. Change the first line to the following:

```
<%@ include file="swcd/swcd.jsp" %>
```

- b. Save the change

▼ How to Configure the Load-Balancing JSP Technology Page for Use With My Desktop

See [“Using My Desktop” on page 175](#) for details of the My Desktop features.

All the SGD servers in the array must be configured in the same way.

1. Log in as superuser (root) on the SGD host.
2. Copy the load-balancing JSP technology page files to the `/sgd/mydesktop` web application directory.

For example:

```
# cd /opt/tarantella/webserver/tomcat/tomcat-version/webapps/sgd/  
# cp -rp admin/loaddist/mydesktop/swcd/
```

Note – When you copy the files, use the `-p` option to preserve the file permissions.

3. Configure My Desktop to use the load-balancing JSP technology page.

- a. Rename the My Desktop entry point JSP technology page.

The entry point JSP technology page is `mydesktop/index.jsp`.

For example:

```
# mv mydesktop/index.jsp mydesktop/mydesktop.jsp
```

- b. Create a new entry point JSP technology page for My Desktop.

Create a new JSP technology page file, `mydesktop/index.jsp`, with the following content:

```
<%@ include file="/mydesktop/swcd/swcd.jsp" %>
```

- c. Check the file permissions for the My Desktop JSP technology page files.

```
# chmod root:ttaserv mydesktop/index.jsp mydesktop/mydesktop.jsp
```

4. Edit the load-balancing JSP technology page.

The load-balancing JSP technology page is `mydesktop/swcd/swcd.jsp`.

a. Add the external DNS names of the SGD servers to be load balanced and set the `LBHOST` variable.

- To use the loadbalancing JSP technology page to distribute user sessions, follow the instructions in [Step 3 of “How to Configure the Load-Balancing JSP Technology Page to Distribute User Sessions” on page 355](#).
- To use an external mechanism to distribute user sessions, follow the instructions in [Step 3 of “How to Configure the Load-Balancing JSP Technology Page for an External Load Balancing Mechanism” on page 358](#).

b. Set the `TARGET` variable.

You must change the `TARGET` variable so that it directs users to My Desktop instead of the webtop.

```
var TARGET="/sgd/mydesktop/mydesktop.jsp"
```

c. Save the changes.

Additional Load-Balancing JSP Technology Page Configuration

This section describes the additional configuration that is available for the load-balancing JSP technology page.

Using Another Webtop

The load-balancing JSP technology page connects users to the standard webtop. To use another webtop, for example a customized webtop, amend the following line:

```
var TARGET="/sgd/standard.jsp"
```

Localized Splash Screen

The load-balancing JSP technology page displays a splash screen in English using the images in the `/sgd/swcd/` directory. To display a localized splash screen, change the default location of the splash screen images in the following lines:

```
// ** Location of gif files
<%
// If the gifs are located in the locale dependent resource use the Path below
```



```
String path = getContextPath(request) + "/resources/images/splash/locale=" +  
getBestSupportedLocale(request) + "/";  
// Default location  
//String path = "swcd/";  
%>
```

Other Variables

The following are the other variables used by the load-balancing JSP technology page:

- **SGDLDCookie**

The name of the cookie used for load balancing purposes.

The default is SGD_SWCDCookie.

- **TIMEOUT**

The time in milliseconds the load-balancing JSP technology page waits for a response from the SGD web server on the selected host. If the timeout period elapses, the next host in the list is tried.

The default is 10000 milliseconds.

- **TESTGIF**

The file the load-balancing JSP technology page attempts to get from the SGD web server on the selected host. This is used to check whether the host is available.

The default is /sgd/resources/images/webtop/secure.gif.

Application Session Load Balancing

Application session load balancing is concerned with choosing an SGD server to manage an application session.

An application session consists of a set of data about the session, for example the user identity of the user that started the session, and a Protocol Engine process. The Protocol Engine process runs on an SGD server and performs the following tasks:

- Maintains the connection to the application on the application server
- Stores the display data for the application
- Sends and receives data over the AIP connection to the client device

A Protocol Engine process can run on any SGD server in the array. This is not necessarily the same server that hosts the user session, which is the SGD server the user logged in to.

SGD can load balance application sessions across all the SGD servers in the array. The more servers you have, the less the load on each. In the Administration Console, you configure application session load balancing on the Global Settings → Performance tab. You can configure SGD to use one of the following methods for selecting the SGD server to host the application session:

- **Server Hosting the User Session** – The SGD server that is hosting the user session
- **Least CPU Usage** – The SGD server with the most central processing unit (CPU) idle time
- **Fewest Application Sessions** – The SGD server that is hosting the fewest application sessions

By default, SGD load balances application sessions using the server that hosts the user session.

Application Load Balancing

Application load balancing is concerned with the following:

- Choosing an application server to run an application so that users get the best performance
- Distributing application starts so that the application servers achieve a similar relative workload

SGD Administrators manage application load balancing by defining the application servers that can run an application, and by selecting the load balancing method to use.

Defining the Application Servers to Run the Application

You define the application servers that can run an application by assigning application server objects to the application object.

In the Administration Console, you do this on the Hosting Application Servers tab for the application. Alternatively, you can assign an application to an application server. You do this on the Hosted Applications tab for the application server object.

You can also assign groups of applications to an application server, or groups of application servers to an application. Groups are useful for creating pools of application servers, sometimes called application server farms, or applications.

In the Administration Console, you can also select and deselect the Application Start check box on the General tab for an application server object. This marks an application server as available or unavailable to run applications. This is useful, for example, to make a server temporarily unavailable during maintenance work.

Selecting the Load Balancing Method

You can select the load balancing method that SGD uses to determine the most suitable application server for the user.

In the Administration Console, you configure a default load balancing method on the Global Settings → Performance tab. You can override the global load balancing method for an individual application by selecting a different method on the Performance tab for the application object.

By default, SGD uses a method that load balances applications by counting the number of application sessions each server is hosting through SGD and then selecting the server with the fewest sessions. SGD also provides methods for load balancing applications based on the *true load of the application server* when a user starts an application. This is called Advanced Load Management. To use Advanced Load Management, you must install the SGD Enhancement Module on every application server.

See [“How Application Load Balancing Works” on page 364](#) for details on how the load balancing method and other factors affect load balancing.

Load Balancing Groups

SGD uses load balancing groups to ensure that connections between SGD servers and application servers take place over high-speed links.

SGD’s Protocol Engines convert the native protocol, such as X11, which is used between the application server and the SGD server, into AIP, which is used between the SGD server and the client device. AIP is optimized for lower bandwidths, but native protocols are not.

If your network includes slow links, you can improve the performance of applications by using load balancing groups. You use load balancing groups to group SGD servers and application servers together. When a user runs an application, SGD tries to ensure that the Protocol Engine process runs on an SGD server in the same group as the application server. This works best when all the application servers and SGD servers in a group are connected by high-speed links.

In the Administration Console, you define the load balancing groups on the Performance tab for an SGD server or an application server. The load balancing group name is simply a string or a comma-separated list of strings. The name can be anything, for example the location in the world or a building code.

How Application Load Balancing Works

The purpose of application load balancing is to select the application server that gives the user the best performance for a particular application. When starting an application, SGD builds a list of candidate application servers using the application servers listed on the Hosting Application Servers tab for the application object. SGD then has to determine which of the candidates is the best one for the user. The decision takes into account the following factors:

- Dynamic application servers
- Application server availability
- Application server filters
- Load balancing groups
- Server affinity
- The relative power of the application servers
- The application server with the least load

The following sections describe how these factors, and your SGD configuration, affect the choice of application server.

Dynamic Application Servers and Load Balancing

If a dynamic application server is configured for the application, the normal SGD mechanisms for application load balancing are overridden because users can choose where to run an application. See [“Dynamic Application Servers” on page 169](#) for more details.

When dynamic application servers are used, only the following *application server attributes* have an effect on the list of application servers presented to a user:

- Application Start (`--available`) – See [“Application Server Availability” on page 365](#) for details
- User Assignment (`--userassign`) – See [“Application Server Filters” on page 365](#) for details
- Maximum Count (`--maxcount`) – See [“Application Server Filters” on page 365](#) for details

Application Server Availability

When starting an application, SGD checks its list of candidate application servers to see if any of them are currently unavailable. If an application server is unavailable, it is removed from the list.

You can use the Application Start (`--available false`) attribute on an application server objects to mark an application server as being unavailable. You can do this, for example, to make an application server unavailable during maintenance work.

If you are using SGD Advanced Load Management, the load balancing service sends regular keep alive packets to SGD. If these packets stop, SGD considers the application server to be “out of contact” and treats the server as unavailable until the load balancing service makes contact again.

Application Server Filters

Application server objects have a Maximum Count (`--maxcount`) and a User Assignment (`--userassign`) attribute that can be used to filter the application servers that run an application for a user.

The Maximum Count attribute is used to limit the number of applications that can run on an application server. Once the limit is reached, SGD does not select the application server to run any more applications. For example, if you assign a group of application servers to an application, and the Maximum Count setting for each application server object is 1, SGD runs each instance of the application on a different application server, and each application server is only used once.

The User Assignment attribute is used to filter application servers based on the user identity (the fully-qualified user name) of the user. It restricts the usage of an application server to particular users or members of LDAP groups. The search filter can be any of the following:

- An RFC2254-compliant LDAP search filter
- An RFC1959-compliant LDAP URL
- A `scottasessionowner=` filter

Note – SGD applies the LDAP-based search filters even if the user identity is not an LDAP identity.

For more information on how to configure User Assignment search filters, see [“User Assignment” on page 650](#).

The Maximum Count and User Assignment attributes can be used individually or together to control the application servers that can run an application for a user. The following table shows the effect of using these attributes on the choice of application server.

| User Assignment | Maximum Count | Effect |
|--------------------------|-----------------|---|
| None | Not set | The application server can run an unlimited number of applications for all users. |
| None | Set to <i>N</i> | The application server can only run <i>N</i> application instances for all users. |
| User search filter | Not set | The application server can run an unlimited number of applications only for the users that match the search filter. |
| User search filter | Set to <i>N</i> | The application server can only run <i>N</i> application instances only for users that match the search filter. |
| LDAP group search filter | Not set | The application server can run an unlimited number of applications only for members of the LDAP group. |
| LDAP group search filter | Set to <i>N</i> | The application server can only run <i>N</i> application instances only for members of the LDAP group. |

Load Balancing Groups

Load balancing groups are used to group SGD servers and application servers together. When a user runs an application, SGD tries to ensure that the Protocol Engine process runs on an SGD server in the same load balancing group as the application server. This works best when all the application servers and SGD servers in a group are connected by high-speed links.

See [“Load Balancing Groups” on page 363](#) for more detail.

Server Affinity

When starting an application, SGD takes into account whether the user is already running any applications on an application server. This is known as *server affinity*. Server affinity means that, if possible, SGD runs an application on the same application server as the last application started by the user.

Note – For server affinity to work efficiently, the applications must be associated with the same set of application servers.

Server affinity is expressed as a percentage. Currently only the following two values are allowed:

- 0 – Any running applications do not affect the choice of application server.
- 100 – Any existing application servers must be reused if they can run the selected application. This is the default value.

You change the server affinity value by running the following command:

```
$ tarantella config edit \  
--tarantella-config-applaunch-appserveraffinity 0|100
```



Caution – If you are using Windows applications, it is best not to change this value, as using multiple application servers causes problems, especially with roaming profiles. There might also be licensing implications for running different applications in a suite of applications on different servers.

The Relative Power of the Application Servers

SGD allows you to factor in the relative power of the application servers to the decision as to where to start an application.

The relative power is expressed as a percentage and by default all servers have a value of 100. By editing the weighting load balancing property for your servers, you can increase or decrease these weightings to increase or decrease the likelihood of SGD choosing an application server. For more details about weightings, see [“Tuning Application Load Balancing” on page 372](#).

You can use the relative power of application servers to do the following:

- Reduce the number of application sessions that are started on a particular server, for example, because it is used for other processes outside of SGD
- Increase the number of application sessions that are started on a particular server, for example, because, although it has less CPU capacity, it has better Input/Output (I/O) capabilities

For more details on how the weighting is used, see the load calculations in [“The Application Server With the Least Load” on page 368](#).

Example Relative Power Calculation 1

You have two application servers, london and paris. Paris has a weighting of 50 and london has a weighting of 100. If all the other conditions for starting applications are met and the servers currently have the same load, london is more likely to be chosen to run the application.

Example Relative Power Calculation 2

You have 100 application servers and you want to make just one of them “more powerful” than the others. Increase the weighting of that server to 200.

The Application Server With the Least Load

SGD supports several methods for selecting the application server with the least load.

You set a default method on the Global Settings → Performance tab in the Administration Console. You can override the default by specifying a different method on the Performance tab for the application object. This allows you to load balance applications in different ways.

The following are the supported application load balancing methods:

- Fewest Application Sessions
- Least CPU Usage
- Most Free Memory

The Least CPU Usage and Most Free Memory methods calculate the *true load of the application server* when a user starts an application. This is called Advanced Load Management. See [“How Advanced Load Management Works” on page 371](#) for more details.

Fewest Application Sessions

The Fewest Application Sessions method allows SGD to choose the application server which is currently running the fewest number of application sessions. It is based on a simple count of the number of application sessions hosted through SGD.

This method is the default.

If you use Advanced Load Management, the Fewest Application Sessions method is used as a fallback whenever there is a problem, for example if the application server load information is not available to the array when the application is started. This might happen, for example, if the primary SGD server is being restarted.

The application server load is calculated using the following formula:

| |
|--|
| $\text{number of application sessions} \times 100 / \text{server weighting}$ |
|--|

Example Load Calculation Using Fewest Application Sessions

The following is an example of how SGD calculates the load using the Fewest Application Sessions method of application load balancing.

The application server london is currently hosting 10 application sessions and has a server weighting value of 100.

The application server paris is currently hosting 12 application sessions, and has a server weighting value of 100.

The load value for london is:

$$10 \times 100/100 = 10$$

The load value for paris is:

$$12 \times 100/100 = 12$$

Assuming the other conditions for starting an application are met, SGD chooses london to run the next two application sessions. If the server weighting value for london was decreased to 50, SGD chooses paris to run the next 8 application sessions, because london's load is now 20 ($10 \times 100/50$).

Least CPU Usage

The Least CPU Usage method allows SGD to choose the application server with the most CPU idle and is suitable for applications that require many processor cycles.

The method measures the application server's load in terms of its CPU capabilities, measured in Bogomips, and by how much of its CPU is being used. These measurements are taken by the load balancing service.

The spare capacity is calculated using the following formula:

$$(\text{Bogomips} \times \text{CPU idle \%}) \times \text{weighting} / 100$$

Example Load Calculation Using Least CPU Usage

The following is an example of how SGD calculates the load using the Least CPU Usage method of application load balancing.

The application server london has a Bogomips measurement of 500, a server weighting value of 75 and has 25% CPU idle.

The application server paris has a BogoMips measurement of 100, a server weighting value of 100 and has 50% CPU idle.

The spare capacity for london is:

$$(500 \times 25) \times 75/100 = 9375$$

The spare capacity for paris is:

$$(100 \times 50) \times 100/100 = 5000$$

Assuming the other conditions for starting an application are met, london is chosen as the application server, even though paris is using less of its CPU and has a higher server weighting value.

Most Free Memory

The Most Free Memory method allows SGD to choose the application server with most free virtual memory and is suitable for applications that require a lot of memory.

The method measures the application server's load by comparing the application server's actual virtual memory with the amount of memory that is currently being used. These measurements are taken by the load balancing service.

The spare capacity is calculated using the following formula:

$$\text{virtual memory free} \times \text{weighting} / 100$$

Example Load Calculation Using Most Free Memory

The following is an example of how SGD calculates the load using the Most Free Memory method of application load balancing.

The application server london has a server weighting value of 100 and has 250 megabytes virtual memory free.

The application server paris has a server weighting value of 75 and has 500 megabytes virtual memory free.

The spare capacity value for london is:

$$250 \times 100/100 = 250$$

The spare capacity value for paris is:

$$500 \times 75/100 = 375$$

Assuming the other conditions for starting an application are met, paris is the chosen application server.

How Advanced Load Management Works

Advanced Load Management enables you to load balance applications based on either the amount of free memory, or the amount of free CPU, the application server has when the application is started. You can only load balance X applications, Windows applications, and character applications using these methods.

To use Advanced Load Management, you must install the SGD Enhancement Module on every application server. This installs a load balancing service which provides SGD with real-time information about the application server's CPU and memory load. It also helps SGD to detect if an application server is unavailable, for example because it is being rebooted.

The following is an overview of how the load balancing service works:

1. Whenever the primary SGD server is started, it builds a list of application servers that need to be considered for load balancing. The list is updated whenever a host is assigned to, or removed from, an application.
2. The primary SGD server contacts each of the load-balanced application servers and requests initial load information. It does this by contacting the load balancing service on TCP port 3579 on each application server. Establishing contact also confirms that the application server is available to run applications.
3. The primary SGD server sends an update to the secondary servers in the array. The update contains capacity values for each of the methods and information about the application servers that are not available.
4. The load balancing service sends regular updates to the primary SGD server on User Datagram Protocol (UDP) port 3579. The updates take place even if the load does not change. The absence of a regular update helps SGD to detect whether an application server is available to run applications.
5. The primary SGD server sends regular updates to the secondary servers in the array. The update contains capacity values for each of the methods and information about the application servers that are not available. The updates take place even if the load does not change.

Note – The load balancing service always sends application server load data to the primary SGD server. If the primary server is not available, Advanced Load Management is not available and the secondary servers revert to the default session-based load balancing instead.

6. The primary or secondary SGD servers starts applications on the basis of the load information they receive in the updates.

Tuning Application Load Balancing

SGD Administrators can tune application load balancing by editing application load balancing properties. These properties affect how the load balancing service used for Advanced Load Management operates, and how SGD calculates the application server load. You can tune application load balancing globally and for individual application servers. See [“Editing Application Load Balancing Properties” on page 375](#) for details on how to edit the load balancing properties.

Before you tune application load balancing, make sure you have read and understood the following:

- [“How Application Load Balancing Works” on page 364](#)
- [“How Advanced Load Management Works” on page 371](#)

You can tune the following aspects of how application load balancing works:

- Application server’s relative power
- Load balancing listening ports
- SGD server requests updates from an application server
- Frequency of the load calculation
- Frequency of updates to the primary SGD server
- Reliability of CPU and memory data
- Frequency of updates to array members

This tuning is described in the following sections.



Caution – With the exception of tuning an application server’s relative power, this tuning only applies if you are using Advanced Load Management.

Application Server's Relative Power

The `weighting` property allows you to factor in the relative power of the application servers to the decision SGD takes as to where to run an application. This is discussed in [“The Relative Power of the Application Servers” on page 367](#).

Load Balancing Listening Ports

The primary SGD server in the array contacts the SGD load balancing service on an application server on TCP port 3579. This is controlled by the `listeningport` property.

The load balancing service sends updates to the primary SGD server on UDP port 3579. This is controlled by the `probe.listeningport` property.

These ports are registered with the Internet Assigned Numbers Authority (IANA) and are reserved for use only by SGD. Only change these properties if Oracle Support asks you to. You must open these ports if you have a firewall between the primary SGD server and the application servers.

SGD Requests Updates From an Application Server

The `connectretries` property is the number of times the primary SGD server tries to connect to an application server to request load updates. The interval between each attempt is controlled by the `shorttimeout` property. If the attempts to connect fail, the SGD server waits for the period of time controlled by the `longtimeout` property before trying again.

For example, using the defaults for these properties, the primary SGD server makes 5 attempts (`connectretries`) to contact the application server at 20 second intervals (`shorttimeout`). If all 5 attempts fail, SGD waits 600 seconds (`longtimeout`) before making 5 more attempts at 20 second intervals.

You might want to change the timeout properties, for example, if an application server takes a long time to reboot.

The `scaninterval` property controls the period of time between scans of the SGD server's list of load-balanced application servers. The scan checks for the application servers that need to be contacted to request a load update (`connectretries`).

The `sockettimeout` property controls how long it is before an SGD server gets an error by trying to contact the load balancing service when there is no data available.

Frequency of the Load Calculation

The `probe.samplerate` and `probe.windowsize` properties control how often the load balancing service measures the application server's average load.

For example, the `probe.samplerate` is 10 seconds and the `probe.windowsize` is 5. After 50 seconds (5×10), the 5 measurements needed to calculate the average have been taken. After a further 10 seconds, the load balancing service takes a new measurement, discards the oldest measurement and then calculates a new average load.

You can increase and decrease the frequency of the calculation depending on how often you expect the application server load to change. For example, if users start applications at the start of the day and then close them at the end of the day, you might want to decrease the frequency of the load calculation. However, if users repeatedly start and stop applications, you might want to increase the frequency of the load calculation.

Frequency of Updates to the Primary SGD Server

The `replyfrequency` property controls the interval at which the load balancing service send updates to the primary SGD server.

The `percentagechange` property controls the minimum percentage change in CPU or memory use that must be reported to the primary SGD server. The load balancing service sends these updates as soon as the percentage change occurs. For example if an application server is running at 30% CPU load and the `percentagechange` value is 10, an update occurs when the load is either 20% or 40%. The load balancing service takes into account sudden "spikes" of activity and also makes adjustments when, for example a server reaches 81% CPU load and the `percentagechange` value is 20%.

The `replyfrequency` updates are sent even if the load does not change and even if there has been a `percentagechange` update. The basis for the percentage change calculation is reset every time there is a `replyfrequency` update.

If there is no update from an application server for `updatelimit x replyfrequency` seconds, SGD considers the application server to be "out of contact". This means the application server is marked as unavailable to run applications until the SGD server can re-establish contact with it.

Reliability of CPU and Memory Data

SGD considers the CPU and memory data it receives to be too unreliable if there has been no update from the application server for `updatelimit x replyfrequency` seconds.

Note – The load balancing service sends updates even if the load does not change.

If the data is unreliable, the data is ignored when making the decision on where to run an application. The net effect of this is to make the application server the last in the queue so that it can only be used to run applications if no other server is available or suitable.

Frequency of Updates to Array Members

The primary SGD server sends CPU and memory load updates to the other members of the array every $\text{maxmissedsamples} \times \text{replyfrequency} / 2$ seconds. This update takes place even if the load does not change.

If a secondary SGD server misses an update, it considers the load data it has to be unreliable and reverts to the Fewest Application Sessions method of load balancing. It uses this method until it receives a new update.

Editing Application Load Balancing Properties

You tune SGD application load balancing by editing application server load balancing properties. The properties are stored in a properties file, which you can edit with a text editor. There are three properties files, as follows:

- **Global properties file** – This file contains the default settings for all the SGD servers in an array
- **Application server properties file** – This file allows you to override some of the default settings in the global properties file for a particular application server
- **Load balancing service properties file** – This file contains the settings the load balancing service uses when it is first started or restarted on a UNIX® or Linux platform application server

This section describes how you edit the properties files and what properties are available. For detailed information on how to use the properties, see [“Tuning Application Load Balancing” on page 372](#).



Caution – Edit these properties with care as it can cause applications to fail to start.

The Global Load Balancing Properties File

The global load balancing properties file contains the default load balancing properties for all the SGD servers in an array.

The file is

`/opt/tarantella/var/serverconfig/global/tier3lb.properties`.



Caution – Only edit these properties on the primary SGD server in the array. The primary copies the amended properties file to the secondary servers.

In the `tier3lb.properties` properties file, the properties are prefixed with `tarantella.config.tier3lb`, for example `tarantella.config.tier3lb.weighting`.

The following table lists the properties you can change, and gives the default value of the property when SGD is first installed. The table also explains what each property is used for.

| Property | Default Value | Purpose |
|-----------------------------------|---------------|--|
| <code>connectretries</code> | 3 | The number of times the SGD server tries to connect to the application server to request CPU and memory usage updates. |
| <code>listeningport</code> | 3579 | The UDP port the SGD server uses to listen for data sent by the load balancing service. |
| <code>longtimeout</code> | 900 | The pause in seconds between groups of attempts by the SGD server to connect to the application server. |
| <code>maxmissedsamples</code> | 20 | The number of missed samples used to calculate whether the CPU and memory data for the application server is too unreliable to be used. |
| <code>probe.listeningport</code> | 3579 | The TCP port the load balancing service uses to listen for requests from SGD servers, for example, when to start sending updates. |
| <code>probe.percentchange</code> | 10 | The minimum percentage increase or decrease in CPU and memory use that must be reported to the SGD server. |
| <code>probe.replyfrequency</code> | 30 | The interval in seconds at which the load balancing service sends CPU and memory measurements to the SGD server. The minimum value for this property is 2. |
| <code>probe.samplerate</code> | 15 | The interval in seconds between CPU and memory measurements. The minimum value for this property is 1. |
| <code>probe.windowsize</code> | 3 | The number of CPU and memory measurements used to calculate the CPU and memory average. The minimum value for this property is 1. |

| Property | Default Value | Purpose |
|---------------|---------------|--|
| scaninterval | 60 | The interval in seconds between scans of the SGD server's list of load-balanced application servers. |
| shorttimeout | 60 | The interval in seconds between attempts by the SGD server to connect to the application server. |
| sockettimeout | 5 | The socket timeout in seconds. |
| updatelimit | 5 | The limit used to calculate when the load balancing service has stopped sending updates. |
| weighting | 100 | The weighting of load measurements relative to the other application servers. |

The following properties also appear in the `tier3lb.properties` properties file, but they must not be changed:

```
tarantella.config.type=server
tarantella.config.name=tier3lb
```

The Application Server Load Balancing Properties File

You can override some of the global load balancing properties by creating a load balancing properties file for a particular application server. You have to *manually* create this file as described in [“How to Create an Application Server Load Balancing Properties File” on page 378](#).

The global properties you can override are the following:

- `probe.listeningport`
- `probe.percentchange`
- `probe.replyfrequency`
- `probe.samplerate`
- `probe.windowsize`
- `weighting`

In the server-specific properties file, the properties are prefixed with `tarantella.config.tier3hostdata`, for example `tarantella.config.tier3hostdata.weighting`.

▼ How to Create an Application Server Load Balancing Properties File

Ensure that no users are logged in to the SGD server, and that there are no application sessions, including suspended application sessions, running on the SGD server.

1. Log in as superuser (root) on the primary SGD server.



Caution – Only create the load balancing properties file on the primary SGD server in the array. The primary copies the file to the secondary servers.

2. Change to the `/opt/tarantella/var/serverconfig/global/t3hostdata` directory.

3. Create the load balancing properties file.

Copy the `template.properties` file to a file called `hostname.properties` in the same directory, where `hostname` is the name of the application server, for example, `paris.indigo-insurance.com.properties`.

4. Edit the load balancing properties file.

- a. Open the properties file in a text editor.

- b. Add the fully qualified name of the application server.

Find the line containing the `tarantella.config.tier3hostdata.name` property.

After the "=", type the fully qualified name of the application server.

Enclose the name in quotes and escape each part of the host name with a backslash. For example:

```
".../_ens/o\=Indigo Insurance/cn\=paris"
```

- c. Configure the server-specific properties.

Uncomment the lines, by deleting the "#", that contain the properties you want to be override.

Only uncomment the properties that you want to be different from the global defaults.

Change the values of the properties you want to override.

Tip – The `template.properties` file contains comments to help you create a server-specific file.

- d. Save the changes and close the file.

5. Do a warm restart of the primary SGD server.

```
# tarantella restart sgd --warm
```

The Load Balancing Service Properties File

The load balancing service properties file contains the settings that are used when the load balancing service is first started, or whenever the service is restarted, on a UNIX or Linux platform application server.

Caution – Only make changes to these properties if you have been asked to by Oracle Support, or if you change the physical or virtual memory of the application server and you have not reinstalled the SGD Enhancement Module.

The load balancing services properties file is:

```
/opt/tta_tem/var/serverconfig/local/tier3loadbalancing.properties.
```

If you change these properties, you must manually stop and restart the load balancing service.

The properties you can override are the following:

- `probe.listeningport`
- `probe.percentchange`
- `probe.replyfrequency`
- `probe.samplerate`
- `probe.windowsize`
- `weighting`

In the load balancing service properties file, the properties are prefixed with `tarantella.config.tier3loadbalancing`, for example `tarantella.config.tier3loadbalancing.weighting`.

SGD Web Server and Administration Console

This section contains information about the web server that is included with SGD and the SGD Administration Console..

This section includes the following topics:

- “Introducing the SGD Web Server” on page 380
- “Securing the SGD Web Server” on page 380
- “Using the Administration Console” on page 381
- “Administration Console Configuration Settings” on page 383
- “Securing Access to the Administration Console” on page 385

Introducing the SGD Web Server

When you install SGD, the SGD web server is also installed. The SGD web server is preconfigured for use with SGD. The components included with the SGD web server are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.

If you have an existing web server on the SGD host, this is not affected by the SGD web server, as the SGD web server listens on a different port.

You can configure the SGD web server using standard Apache directives. See the [Apache documentation](#) for details.

You can control the SGD web server independently of the SGD server, using the `tarantella start webserver`, `tarantella stop webserver`, and `tarantella restart webserver` commands.

Securing the SGD Web Server

By default, the SGD web server is configured to be a secure HTTPS web server and to share the SGD server SSL certificate used for SGD security services.

Directory indexes are disabled by default for the SGD web server. This means that users cannot browse the directories on the SGD web server.

If you require enhanced security, a more secure version of the `httpd.conf` Apache configuration file used by the SGD web server is supplied. See “[The httpd.conf.secure File](#)” on page 381 for more details about this file.

The httpd.conf.secure File

The `httpd.conf.secure` file is an Apache server configuration file that configures the SGD web server for enhanced security. The file is included with the SGD distribution, at `/opt/tarantella/webserver/apache/apache-version/conf/httpd.conf.secure` on the SGD host.

The `httpd.conf.secure` file provides the following additional security features, compared to the standard `httpd.conf` file used by the SGD web server:

- Apache modules that are not used by SGD are disabled
- Access to the `/cgi-bin` directory on the SGD web server is not allowed

To use `httpd.conf.secure` on an SGD server that has previously been secured, you must first disable security on the SGD server, before installing the `httpd.conf.secure` file. You can then enable security services for the SGD server, as described in “[Secure Connections to SGD Servers](#)” on page 18.



Caution – Do not use `httpd.conf.secure` if you have used the `tarantella security enable` command to configure security automatically on the SGD server.

Using the Administration Console

This section describes how to run the Administration Console. It also includes details of how to avoid some common problems when using the Administration Console.

Supported Browsers for the Administration Console

To display the Administration Console, you can use a supported browser. The browser must have the JavaScript programming language enabled. The supported browsers are listed in the *Oracle Secure Global Desktop 4.6 Platform Support and Release Notes* available at <http://docs.sun.com/app/docs/doc/821-1928>.



Caution – When using the Administration Console, do not use the browser’s Back button. Instead, use the Jump to Object View and Jump to Navigation View links, or the Object History list, to navigate through the Administration Console pages.

Starting the Administration Console

The Administration Console works best when you run it on the primary SGD server in the array.

You can start the Administration Console in the following ways:

- Click the Administration Console link on the webtop of an SGD Administrator.
- Click the Launch the Oracle Secure Global Desktop Administration Console link on the SGD web server Welcome Page at `http://server.example.com`, where *server.example.com* is the name of an SGD server.
- Go to the `http://server.example.com/sgdadmin` URL.

Note – The Administration Console is for SGD Administrators only. To use the Administration Console you must log in as, or be logged in as, an SGD Administrator.

Deploying the Administration Console on Other Web Application Containers

The Administration Console is only supported when used with the SGD web server.

The Administration Console ships with a web application archive (WAR) file, `sgdadmin.war`. Using this file to redeploy the Administration Console on another web application server is not supported.

Avoiding SGD Datastore Update Problems

You can perform operations on the SGD datastore, such as creating new objects and editing object attributes, using the Administration Console from any SGD server in the array.

When you edit the SGD datastore, the changes you make are sent to the primary SGD server. The primary SGD server then replicates these changes to all secondary servers in the array.

By running the Administration Console from the primary SGD server, you can avoid problems due to the following:

- **Slow network.** If the network is slow, “Object not found” or “Object not created” errors can be returned. Also, problems with stale data can occur, where configuration changes are not shown correctly.
- **Primary down.** If the primary server is down, or unavailable, SGD datastore changes are not applied.

Performing Array Operations Using the Administration Console

The following limitations apply when using the Administration Console to perform array operations, such as array joining or array detaching:

- **Use the primary SGD server.** Running the Administration Console on the primary server avoids data replication problems. See also [“Avoiding SGD Datastore Update Problems” on page 382](#).
- **All servers involved in an array operation must be up.** For example, you cannot use the Administration Console to detach a secondary server that is down. Instead, use the `tarantella array detach` command.
- **Clocks must be synchronized on all servers involved in an array operation.** For example, you cannot add a secondary server if its clock is out of synchronization by more than a minute. You can use NTP software or the `rdate` command to ensure the clocks on all SGD hosts are synchronized.

Administration Console Configuration Settings

The deployment descriptor for the Administration Console web application contains settings that control the operation of the Administration Console. The deployment descriptor is the following file:

```
/opt/tarantella/webserver/tomcat/tomcat-version/sgdadmin/WEB-INF/web.xml
```

This section describes the settings in the deployment descriptor that you might want to configure. Most of the settings are context parameters, contained in `<context-param>` elements. You must not change any other settings in the `web.xml` file.

When working with deployment descriptor settings, note the following:

- Only change `web.xml` if you understand what you are doing.
- Always create and keep a backup of the original `web.xml`, in case you need to revert to a previous version. See [“Backing Up and Restoring an SGD Installation” on page 414](#) for advice on how to do this.
- After changing `web.xml`, you must always restart the SGD web server for the changes to take effect.
- Changes to `web.xml` only apply for the server that is hosting the Administration Console.
- You must not change the order of the Extensible Markup Language (XML) elements contained in `web.xml`.

Number of Search Results

The `com.sun.tta.confmgr.DisplayLimit` context parameter enables you to configure the maximum number of search results you can display in the Administration Console. The default is 150. If there are more results than the display limit, the Administration Console displays a message. Increasing the display limit can have an effect on performance. Set the display limit to 0 to see unlimited search results.

Synchronization Wait Period

The `com.sun.tta.confmgr.ArraySyncPeriod` context parameter is only used if you are running the Administration Console from a secondary server, and you create or edit objects in the SGD datastore. This parameter enables you to configure the period of time, in milliseconds, that the Administration Console waits for changes to be copied across the array before proceeding. The default value is 250. The Administration Console waits for twice this setting, a default of 0.5 seconds, before proceeding.

Searching and Displaying LDAP Data

The `com.sun.tta.confmgr.LdapSearchTimeLimit` context parameter enables you to configure the maximum time, in milliseconds, to allow for a search of a Lightweight Directory Access Protocol (LDAP) directory. The default is 0, which means the search time is unlimited. Only change this context parameter if you have particularly slow LDAP directory servers.

The following context parameters are used to filter the display of LDAP data, when you select Local + LDAP in the Repository list in the Administration Console:

- Filters used by the navigation tree. These are the following context parameters:
 - `com.sun.tta.confmgr.LdapContainerFilter`
 - `com.sun.tta.confmgr.LdapUserFilter`
 - `com.sun.tta.confmgr.LdapGroupFilter`
- Filters used when you search an LDAP directory. These are the following context parameters:
 - `com.sun.tta.confmgr.LdapContainerSearchFilter`
 - `com.sun.tta.confmgr.LdapUserSearchFilter`
 - `com.sun.tta.confmgr.LdapGroupSearchFilter`
- Filters used when you load the LDAP assignments on the Assigned Applications tab for a user profile. This is the `com.sun.tta.confmgr.LdapMemberFilter` context parameter.

These context parameters contain the definitions of what the Administration Console considers as LDAP containers, users, and groups. You might want to change these filters to improve performance, or to change the definition of these LDAP object types to match those used in your LDAP directory. To avoid inconsistencies, if you change a filter for the navigation tree, you must also change the filter used for the LDAP search.

Session Timeout

The `session-timeout` setting defines the period of time after which the user is logged out if there is no activity, meaning no HTTP requests, in the Administration Console. The default setting is 30 minutes, to ensure unattended Administration Console sessions are not left open indefinitely.

Note – The `session-timeout` setting is independent of the timeout attribute for inactive user sessions, `tarantella-config-array-webtopsessionidletimeout`.

Securing Access to the Administration Console

Because the Administration Console is a web application, you can control which client devices are allowed to access it.

For example, you can do this by configuring the SGD web server to use the Apache `<Location>` directive, as in the following example:

```
<Location /sgdadmin>
    Order Deny,Allow
    Deny from all
    Allow from 129.156.4.240
</Location>
```

In this example, only client devices with an IP address of 129.156.4.240 are allowed to access the `/sgdadmin` directory on the SGD web server. The `/sgdadmin` directory contains the home page of the Administration Console.

For more information on how to configure the `<Location>` directive, check the [Apache documentation](#).

Monitoring and Logging

This section describes the SGD datastore, how to monitor user activity, and how to configure logging.

This section includes the following topics:

- [“The SGD Datastore” on page 386](#)
- [“User Sessions and Application Sessions” on page 387](#)
- [“Using Log Filters to Troubleshoot Problems With an SGD Server” on page 391](#)
- [“Using Log Filters for Auditing” on page 398](#)
- [“Using Log Filters to Troubleshoot Problems With Protocol Engines” on page 402](#)
- [“SGD Web Server Logging” on page 406](#)
- [“SGD Client Logging” on page 407](#)

The SGD Datastore

The SGD servers in an array share information. Each SGD server knows:

- What applications are configured and the applications servers on which they run
- Which users have access to applications
- Who is logged in to SGD
- Which applications users are running

The collection of information is known as *the datastore*.

Information about users, applications, application servers, and webtops is stored on disk in the local repository. Information about user and application sessions is stored in memory.

The datastore is organized into namespaces which are displayed and used on the command line and log files. The general structure is `.../namespace/name-within-namespace`. The `...` indicates the root of the datastore. The namespace indicates the source of the information, such as DNS. After the namespace, names are written using whichever naming scheme is appropriate to the namespace. Names are written like file system paths (slash-separated and top-down).

The following are some commonly used namespaces.

| Namespace | Description | Example |
|-----------|-------------------------------------|--|
| Local | SGD objects in the local repository | .../_ens/o=Indigo Insurance/ou=Marketing/cn=Cust-o-Dat |
| LDAP | Objects in an LDAP directory | .../_service/sco/tta/ldapcache/cn=Cust-o-Dat,ou=Marketing,o=Indigo Insurance |
| DNS | Machines on the network | .../_dns/verona.indigo-insurance.com |

User Sessions and Application Sessions

This section describes the differences between user sessions and application sessions in SGD. Using the Administration Console and the command line to monitor and control user and application sessions is also covered.

This section includes the following topics:

- [“User Sessions” on page 387](#)
- [“Application Sessions” on page 389](#)
- [“Anonymous Users and Shared Users” on page 390](#)

User Sessions

A user session begins when a user logs in to SGD and ends when a user logs out of SGD. User sessions are hosted by the SGD server the user logs in to. The user name and password they type determine the type of user they are. See [Chapter 2](#) for more details about user authentication.

If a user logs in and they already have a user session, the user session is transferred to the new SGD server and the old session ends. This is sometimes called *session moving*, or *session grabbing*.

User sessions can be *standard* sessions or *secure* sessions. Secure sessions are only available when SGD security services are enabled. See [“Secure Connections to SGD Servers” on page 18](#) for more details.

In the Administration Console, you can list user sessions as follows:

- The Sessions tab, in Navigation View, shows all the user sessions that are running on all SGD servers in the array
- The User Sessions tab for an SGD server shows all the user sessions that are hosted by that SGD server

- The User Sessions tab for a user profile shows all the user sessions associated with the user profile

The Sessions tab and User Sessions tab enable you to select and end user sessions. The User Sessions tab enables you to view further details about the user session. For example, the information that the SGD Client detects about the client device.

On the command line, you use the `tarantella webtopsession` command to list and end user sessions.

Idle User Session Timeout

You can configure an idle timeout period for inactive user sessions. User sessions are ended automatically if there has been no activity on the AIP connection between the SGD Client and the SGD server for the specified period. The timeout is disabled by default for an SGD array.

Activity on the following devices has no effect on the idle timeout period:

- Serial ports
- Smart cards
- Audio

To specify the idle timeout attribute, go to the Global Settings → Communication tab of the Administration Console and type a value in the User Session Idle Timeout field.

Alternatively, use the following command:

```
$ tarantella config edit --webtop-session-idle-timeout secs
```

Replace *secs* with the timeout value, measured in seconds. A setting of 0 turns off the user session idle timeout feature. This is the default setting.



Caution – Do not configure an idle timeout that is less than 300 seconds (five minutes).

You must restart every SGD server in the array for changes to this attribute to take effect.

Application Sessions

An application session begins when a user starts an application and ends when the application exits. Each application session corresponds to an application currently running through SGD.

An application session can be hosted by any SGD server in the array. This might not be the same SGD server that the user logged in to, see [“Arrays” on page 333](#).

Each application session has a corresponding Protocol Engine process. The Protocol Engine handles the communication between the client device and the application server. The Protocol Engine also converts the display protocol used by the application to the AIP, which is understood by the SGD Client running on the client device.

You can use application session load balancing to spread the load of the Protocol Engines among the SGD servers in the array. See [“Application Session Load Balancing” on page 361](#) for more details.

Some applications can be configured to keep running, even when they are not displayed. These are called *resumable* applications.

Each application object has an Application Resumability attribute that determines the application’s resumability behavior. Applications can have one of the following Application Resumability settings.

| Setting | Description |
|-------------------------|---|
| Never | The application exits when the user logs out of SGD. You cannot suspend or resume applications that are non-resumable. |
| During the User Session | The application continues to run until the user logs out of SGD. While they are logged in, the user can suspend and resume these applications. |
| General | The application continues to run even after the user has logged out of SGD. When the user logs in again, they can click the Resume button to display the running application again. |

If an application is resumable, it is resumable for a period of time, specified by a timeout. If the SGD Client exits unexpectedly, the timeout period is the configured timeout plus 20 minutes.

Resumable applications are useful for the following reasons:

- Applications that take a long time to start can be left running, even after users have logged out of SGD
- Mobile users can leave applications running while they travel
- Users can easily recover from browser or other crashes

In the Administration Console you can list application sessions as follows:

- The Application Sessions tab for an SGD server shows all the application sessions that are hosted by that server
- The Application Sessions tab for a user profile shows all the application sessions associated with the user profile
- The Application Sessions tab for an application server shows all the applications that are running on that application server

The Applications Sessions tab enables you to view details about each application session. You can also end and shadow application sessions. Shadowing a session enables you and the user see and interact with the application at the same time.

See [“Using Shadowing to Troubleshoot a User’s Problem” on page 214](#) for more details about shadowing application sessions.

Note – You can only shadow Windows applications and X applications. The application sessions must not be suspended.

On the command line, you use the `tarantella emulatorsession` command to list and end user sessions.

Anonymous Users and Shared Users

There are two special cases, as follows:

- **Anonymous users.** These are users that log in without typing a user name and password. See [“Anonymous User Authentication” on page 61](#).
- **Shared users.** Also called *guest users*. These are users that log in with the same user name and password. See [“Using Shared Accounts for Guest Users” on page 105](#).

To be able to distinguish between these users, SGD assigns shared users and anonymous users a temporary user identity when they log in. This has the following effects:

- User sessions do not transfer if the user logs in to SGD more than once
- Application sessions end as soon as the user session ends, regardless of the application’s Application Resumability setting
- If the user does not log out, server resources are consumed

Using Log Filters to Troubleshoot Problems With an SGD Server

When you first install SGD, the default log filters log all errors on the SGD server. If you want to obtain more detailed information, for example to troubleshoot a problem, you can set additional log filters.

You can set additional log filters in the following ways:

- Type the filter in the Log Filter field on the Global Settings → Monitoring tab in the Administration Console. Each filter must be separated by a Return key press.
- Use the following command:

```
$ tarantella config edit --array-logfilter filter...
```

Each *filter* must be separated by a space.

Each log filter has the form:

```
component/subcomponent/severity:destination
```

The options for each part of the filter, and how you view the log output, are described in the following sections.

Note – Log filters can create large amounts of data. It is good practice to set as specific a filter as possible and then remove the filter when you have finished with it.

Selecting a Component and Subcomponent

Selecting a component and subcomponent enables you to choose the area of information you want to log from the SGD server.

The following table shows the available component/subcomponent combinations and an explanation of the kind of information produced.

| Component and Subcomponent | Information Provided |
|----------------------------|--|
| audit/glue | <p>Audit of changes made to the SGD server configuration, or to your local repository configuration, and who made the changes.</p> <p>For example, you can use this to find out who made changes to a user profile object.</p> |
| audit/session | <p>Starting and stopping user sessions and application sessions.</p> <p>For example, you can use this to find out how long a user had a running application session.</p> |
| cdm/audit | <p>Authorization of SGD user for client drive mapping (CDM) purposes.</p> <p>For example, you can use this to find out whether a user's credentials are causing CDM to fail.</p> |
| cdm/server | <p>Information about CDM services.</p> <p>For example, you can use this to find out whether a client connection error is causing CDM to fail.</p> |
| common/config | <p>How SGD server configuration is stored and copied across the array.</p> <p>For example, you can use this to find out why a global setting configuration change is not being applied to an SGD server.</p> |
| metrics/glue | <p>Memory and timings.</p> <p>For example, you can use this to find out how long it took to run an SGD command.</p> |
| metrics/soap | <p>The SOAP component of Tomcat's SOAP proxy.</p> <p>For example, you can use this to trace how long it took a SOAP request to finish.</p> |
| server/billing | <p>SGD billing services.</p> <p>For example, you can use this to find out why billing data is being lost.</p> |
| server/common | <p>General SGD information.</p> <p>For example, you can use this to troubleshoot DNS errors.</p> |
| server/config | <p>Changes to SGD server configuration.</p> <p>For example, you can use this to log changes to SGD server configuration or to find out if the configuration has become corrupt.</p> |

| Component and Subcomponent | Information Provided |
|---------------------------------------|---|
| <code>server/csh</code> | The SGD client session handler. For example, you can use this to find out why a user can not restart an application session. |
| <code>server/deviceservice</code> | Mapping of users to accessible device data. For example, you can use this to find out why a user can not access client drives. |
| <code>server/directoryservices</code> | Connections to Active Directory or LDAP. For example, you can use this to find out why an Active Directory or LDAP user cannot log in. |
| <code>server/diskds</code> | Information about the local repository. For example, you can use this to get information about corrupt objects or inconsistencies in the local repository. |
| <code>server/failover</code> | Information about array failover. For example, you can use this to troubleshoot problems with an SGD array where the primary server is unavailable. |
| <code>server/glue</code> | The protocol used for communication between SGD servers. For example, you can use this to find out why SGD servers cannot communicate. |
| <code>server/install</code> | Installation and upgrades. For example, you can use this to investigate problems with an installation. |
| <code>server/kerberos</code> | Windows Kerberos authentication. For example, you can use this to find out why an Active Directory user cannot log in. |
| <code>server/launch</code> | Starting or resuming applications. For example, you can use this to find out why a user cannot start an application. |
| <code>server/loadbalancing</code> | User session and application load balancing. For example, you can use this to find out why an SGD server is not selected to host application sessions. |
| <code>server/logging</code> | Logging. For example, you can use this to find out why log messages are not being written to a file. |
| <code>server/login</code> | Log in to SGD. For example, you can use this to find out which authentication mechanism authenticated a user and the user profile used. |

| Component and Subcomponent | Information Provided |
|----------------------------|---|
| server/mupp | The SGD MULTiplePlexing Protocol (MUPP) protocol. Only use this filter if Support ask you to. |
| server/printing | SGD printing services. For example, you can use this to find out why print jobs are failing. |
| server/replication | Copying data between SGD servers in an array. For example, you can use this to find out why data has not been copied between array members. |
| server/securid | Connections to SecurID RSA Authentication Manager. For example, you can use this to find out why SecurID authentication is not working. |
| server/security | Secure SSL-based connections. For example, you can use this to find out why the SSL Daemon is not running. |
| server/server | The SGD server component. For example, you can use this to troubleshoot SGD server failures, such as Java™ technology runtime exceptions which are not logged elsewhere. |
| server/services | Internal SGD server services. For example, you can use this to find out why a service is failing. |
| server/session | User sessions. For example, you can use this to find out why a session failed to suspend. |
| server/soap | SOAP bean interface. For example, you can use this to diagnose problems with the SOAP beans. |
| server/soapcommands | SOAP requests. For example, you can use this to log the SOAP requests received. |
| server/tier3loadbalancing | Application server load balancing. For example, you can use this to find out why an application server is not being selected to run an application. |

| Component and Subcomponent | Information Provided |
|----------------------------|---|
| server/tokencache | Authentication token cache. For example, you can use this to find out why an authentication token is not being created for a user. |
| server/tscal | Windows Terminal Services Client Access Licenses (CALs) for non-Windows clients. For example, you can use this to find out why a non-Windows client does not have a CAL. |
| server/webtop | Webtop content, if you are using Directory Services Integration. For example, you can use this to find out why an application is not appearing on a user's webtop. |

Selecting the Severity

You can select one of the following levels of severity for each log filter.

| Severity | Description |
|---------------------------|--|
| <code>fatalerror</code> | Logs information on fatal errors. Fatal errors stop the SGD server from running. When you first install SGD, all fatal errors are logged by default. |
| <code>error</code> | Log information on any errors that occur. When you first install SGD, all errors are logged by default. |
| <code>warningerror</code> | Log information on any warnings that occur, for example if system resources are running low. When you first install SGD, all warnings are logged by default. |
| <code>info</code> | Informational logging. Useful for problem solving and identifying bugs. |
| <code>moreinfo</code> | Verbose informational logging. |
| <code>auditinfo</code> | Logs selected events for auditing purposes, for example changes to SGD server configuration. For details see, “Using Log Filters for Auditing” on page 398 . |

The `fatalerror` severity level produces the least amount of information. The `moreinfo` severity level produces the most information.

Selecting a severity level is not cumulative. For example, selecting `info` does not mean you also see `warningerror` or `fatalerror` log messages.

To log more than one level of severity, use a wild card.

Using Wildcards

You can use a wildcard (*) to match multiple components, subcomponents, and severities.

For example, to log all warning, error, and fatal error messages for printing, you can use a `server/printing/*error` log filter.

Note – If you use a wildcard on the command line, you must enclose the filter in straight quotation marks, to stop your shell from expanding them.

Selecting a Destination

When selecting a destination for the logs, you can specify that the output goes to one of the following destinations:

- A log file
- A log handler

These destinations are described in the following sections.

Using Log Files

If you direct the output to a log file, you can output to the following types of file:

- `filename.log`

SGD formats this log output so that it is easy to read.

This format is required by the `tarantella query errlog` command. This command only searches log files that have names that end `error.log`.

- `filename.jsl`

SGD formats this log output for automated parsing and searching.

This format is required by the `tarantella query audit` command.

The file extension of the destination file controls the format of the file.

You can also create a separate log file for each process ID, by including the `%%PID%%` placeholder in the file name.

The log files are output in the `/opt/tarantella/var/log` directory. You cannot change the location of the log files, but you can use a symbolic link to redirect the logs to a different location. Alternatively, you can use the `syslog` log handler described in [“Using Log Handlers” on page 397](#).

Using Log Handlers

A log handler is a JavaBeans™ component used as the destination for the log messages. When specifying a log handler, you must use its full name. SGD provides the following log handlers:

- **ConsoleSink.** The ConsoleSink log handler writes log messages in a easy-to-read format to standard error. This log handler is enabled by default and logs all fatal errors.

The full name of this log handler is:

```
.../_beans/com.sco.tta.server.log.ConsoleSink
```

- **SyslogSink.** The SyslogSink log handler writes log messages to the UNIX or Linux platform syslog facility.

The full name of this log handler is:

```
.../_beans/com.sco.tta.server.log.SyslogSink
```

Examples of Using Log Filters

The following are some examples of commonly used log filters:

- To debug user logins:

```
server/login/*:login%%PID%%.log  
server/login/*:login%%PID%%.jsl
```

- To troubleshoot CDM:

```
cdm/*/*:cdm%%PID%%.log  
cdm/*/*:cdm%%PID%%.jsl  
server/deviceservice/*:cdm%%PID%%.log  
server/deviceservice/*:cdm%%PID%%.jsl
```

- To troubleshoot printing problems:

```
server/printing/*:print%%PID%%.log  
server/printing/*:print%%PID%%.jsl
```

- To get timing measurements for server performance:

```
metrics/*/*info:metrics.log  
metrics/*/*info:metrics.jsl
```

- To send all warnings, errors, and fatal errors to syslog:

```
*/**error:.../_beans/com.sco.tta.server.log.SyslogSink
```

Viewing Log Output

To view the log output, you can do either of the following:

- Open the `.log` files in a viewer or text editor
- Use the `tarantella query` command

If you use the `tarantella query` command, use the following command options:

- `tarantella query errlog` – To see only the errors and fatal errors for specific SGD server components
- `tarantella query audit` – Searches the logs for any messages relating to a person, an application, or an application server

Note – You can only use these commands to view the log output until the logs are archived. You configure archiving when you install SGD, but you can change the settings at any time by running the `tarantella setup` command.

Using Log Filters for Auditing

SGD enables you to set log filters to provide an audit of the following system events:

- Starting and stopping an SGD server
- Starting and stopping SGD security services
- Changes to object configuration in the local repository
- Changes to global and SGD server configuration
- Unsuccessful attempts to log in to an SGD server
- Logging in and out of SGD
- Starting and stopping an application session

To audit these events, you must set a `*/**auditinfo` log filter.

You can use any of the standard destinations as a destination for the output, but you must direct the output to a `.jsl` file if you want to view the audit information from the command line.

See [“Using Log Filters to Troubleshoot Problems With an SGD Server”](#) on page 391 for more information about setting log filters.

Note – Log output is only created while an SGD server is actually running. If an SGD server is stopped, only the UNIX system root user can perform any of the auditable events.

For each of the events, the log filter records following:

- The date and time of the event
- The type of event
- The result of the event, whether it was successful or it failed
- The identity of the user responsible for the event
- Any other relevant information about the event, for example what was changed and to what value

Viewing Audit Log Information

You can use any of the standard methods for viewing the log output. However, the following command is the most useful:

```
# tarantella query audit --format text|csv|xml --filter "filter"
```

If you select the `text` format, SGD formats the log output so that it is easy to read on screen, but it does not show every detail logged. Using the `csv` format shows every detail logged, but it is only suitable for outputting to a file.

The "*filter*" is an [RFC2254-compliant](#) LDAP search filter. The command searches the log fields in the log files for matching entries to display. For auditing purposes, the most useful log fields are shown in the following table.

| Log Field | Description |
|----------------|--|
| log-category | For auditing purposes, the log-category is always <code>*auditinfo</code> , but this can be any of the standard log filter component/subcomponent/severity settings. |
| log-date | The system date and time when the event took place. The format is <code>yyyy/MM/dd HH:mm:ss.SSS</code> . |
| log-event | The name of the event. |
| log-ip-address | The IP address of a client or server associated with an event. |
| log-keyword | The keyword identifier for the auditable event. |
| log-localhost | The peer DNS name of the SGD server where the event took place. |
| log-pid | The process ID of the event. |

| Log Field | Description |
|-------------------|--|
| log-security-type | The type of security used on a connection, <code>std</code> or <code>ssl</code> . |
| log-systime | The system Coordinated Universal Time (UTC) time, in milliseconds, when the event took place. |
| log-tfn-name | The full name of an object associated with an event. For example, starting an application session might record the name of the user, the application, and the application server. |

Note – A complete list of all the log fields is available in the `/opt/tarantella/var/serverresources/schema/log.at.conf` schema file.

The following table below shows all the log-keywords, and their corresponding log-events, together with a description of the event.

| Log-Keyword | Log-Event | Description |
|---------------|----------------------|--|
| createFailure | createFailure | A user tried to create an object in the local repository but failed. |
| createSuccess | createSuccess | A user created an object in the local repository. |
| deleteFailure | deleteFailure | A user tried to delete an object in the local repository but failed. |
| deleteSuccess | deleteSuccess | A user deleted an object in the local repository. |
| loginFailure | loginResultReconnect | The SGD server requested the client to reconnect on a different port. |
| loginFailure | loginResultFailed | None of the enabled authentication mechanisms authenticated the user. |
| loginFailure | loginResultRejected | User was denied a login by a login filter. For example, this might be because logins are currently not enabled for that particular server, or because the user is currently not allowed to log in. |
| loginFailure | loginResultDisabled | The SGD server is not currently accepting connections. |
| loginFailure | loginResultNoAmbig | An ambiguous login failed because the SGD server does not support ambiguous logins. |
| loginFailure | loginResultAmbiguous | An ambiguous login failed because the user did give enough disambiguation information. |
| loginFailure | loginResultAnonymous | An anonymous login failed because the SGD server does not support anonymous logins. |

| Log-Keyword | Log-Event | Description |
|----------------|-----------------------------|--|
| loginFailure | loginResultNoSecurity | Login failed because the user requires a secure connection, but the connection was made to the standard port. |
| loginFailure | loginResultUnresolveable | Login failed because the SGD server was unable to resolve which user had logged in. |
| loginFailure | loginResultUnknown | Login failed because the SGD server was unable to process an unexpected login result. |
| loginSuccess | webtopSessionStartedDetails | Started a user session for a user. |
| logout | webtopSessionEndedDetails | Stopped a user session for a user. |
| modifyFailure | modifyFailure | A user tried to change an object in the local repository, to change global settings, or to change the configuration of an SGD server but failed. |
| modifySuccess | modifySuccess | A user changed an object in the local repository, changed global settings, or changed the configuration of an SGD server. |
| renameFailure | renameFailure | A user tried to rename an object in the local repository but failed. |
| renameSuccess | renameSuccess | A user renamed an object in the local repository. |
| serverStart | serverStart | The SGD server was started. |
| serverStop | serverStop | The SGD server was stopped. |
| sessionEnded | sessionEndedDetails | Stopped an application session for a user. |
| sessionStarted | sessionStartedDetails | Started application session for a user. |
| sslStart | securitySSLStart | Started SGD security (SSL) services. |
| sslStop | securitySSLStop | Stopped SGD security (SSL) services. |

Examples of Using Log Filters for Auditing

To search for failed log in attempts, use the following filter:

```
--filter "(&(log-category=*auditinfo)(log-keyword=loginFailure))"
```

To search for changes to made to the SGD server configuration by the Administrator Bill Orange, use the following filter:

```
--filter "(&(log-category=*auditinfo)(log-keyword=modifySuccess)(log-tfn-name=.../ens/o=Indigo Insurance/ou=IT/cn=Bill Orange))"
```

Using Log Filters to Troubleshoot Problems With Protocol Engines

When you first install SGD, the default log filters record all errors for protocol engines (PEs). To obtain more information about PE activity, you can set additional PE log filters.

PE log filters can be set for individual PEs, and for the Protocol Engine Manager (PE Manager) process. You configure a PE log filter by setting one of the attributes shown in the following table.

| PE Filter Attribute | Protocol Engine |
|--|---------------------------|
| --tarantella-config-auxserver-logfilter | PE Manager |
| --tarantella-config-execpeconfig-logfilter | Execution Protocol Engine |
| --tarantella-config-xpeconfig-logfilter | X Protocol Engine |
| --tarantella-config-tpeconfig-logfilter | Character Protocol Engine |
| --tarantella-config-ppconfig-logfilter | Print Protocol Engine |
| --tarantella-config-cpeconfig-logfilter | Channel Protocol Engine |

You can only set a PE log filter from the command line. Use the following command:

```
$ tarantella config edit --PE-filter-attribute filter...
```

where *PE-filter-attribute* defines the PE filter attribute you want to configure, and *filter* defines the log filter. For multiple log filter definitions, use straight quotation marks to separate each *filter* parameter.

Each log filter has the form:

```
component/subcomponent/severity
```

The following table shows the available component names for PE logging.

| Protocol Engine | Component |
|---------------------------|--------------|
| PE Manager | pem |
| | proxy |
| Execution Protocol Engine | execpe |
| | launchhelper |
| X Protocol Engine | xpe |
| | pem |

| Protocol Engine | Component |
|---------------------------|-----------|
| Character Protocol Engine | tpe |
| | pem |
| Print Protocol Engine | ppe |
| | pem |
| Channel Protocol Engine | cpe |
| | pem |

For subcomponent, type * to include information on all subcomponents.

You can select the following levels of severity:

- * – Includes all error and warning messages.
- *info – Includes info, moreinfo, and auditinfo messages.
- *error – Includes error, fatalerror, and warningerror messages. This is the default severity.

Changes to the Execution, X, Character, Print, and Channel PE log filters take effect when a new PE is started.

Changes to the PE Manager log filters require an SGD server restart.

Note – Log filters can create large amounts of data. It is good practice to set as specific a filter as possible and then remove the filter after you have finished with it. See [“Resetting a PE Log Filter” on page 405](#) for details of how to do this.

Examples of Using PE Log Filters

The following are some examples of how to use a PE log filter.

- To troubleshoot CDM for X and Windows applications:

```
--tarantella-config-xpeconfig-logfilter "xpe/cdm/*"
```

- To troubleshoot problems with X and Windows applications:

```
--tarantella-config-xpeconfig-logfilter "xpe/*/*" "pem/*/*"
```

- To troubleshoot application launches, you must first enable debugging in the SGD login scripts, as shown in [“An Application Does Not Start” on page 203](#). Then, configure the Execution Protocol Engine log filter as follows:

```
--tarantella-config-execpeconfig-logfilter "execpe/*/*"
```

Note – For the `execpe`, `xpe`, `tpe`, `ppe`, and `cpe` log filters, using the `pem/*` filter shows relevant PE Manager messages for the Protocol Engine.

PE Log File Destinations

PE log files have the file name extension `.log`. SGD formats this type of log output so that it is easy to read.

PE log file names include the name of the PE component and the process ID. For example, messages for a PE Manager running with process ID 4512 are stored to a file named `pemanager4512.log`.

Error messages with a severity of `error`, `fatalerror`, or `warningerror` are stored to a PE log file with a name that ends `error.log`. For example, error messages for a Character Protocol Engine running with process ID 2256 are stored to a file named `cpe2256_error.log`. Files such as this are used by the `tarantella query errlog` command, which only searches log files that have names that end `error.log`.

PE log filter output is directed automatically to log files in the `/opt/tarantella/var/log` directory on the SGD host. You cannot change the location of the log files, but you can use a symbolic link to redirect the logs to a different location.

Viewing PE Log Output

To view the PE logs, do either of the following:

- Open the `.log` files in a viewer or text editor.

On each SGD server in the array, the `.log` files contain messages for PEs running on that specific SGD server.

- Use the `tarantella query errorlog` command to show error messages for PEs.

This command can be used to search all PE error logs in the array.

For example, to display X Protocol Engine error messages for all SGD servers in the array, use the following command:

```
$ tarantella query errlog xpe
```

For example, to display X Protocol Engine error messages for the SGD server `boston.indigo-insurance.com`, use the following command:

```
$ tarantella query errlog xpe --server boston.indigo-insurance.com
```

Note – You can only use these commands to view the log output until the logs are archived. You configure archiving when you install SGD, but you can change the settings at any time by running the `tarantella setup` command.

Resetting a PE Log Filter

As log filters can create large amounts of data, it is good practice to reset the filter to its default configuration after you have finished with it.

The default PE log filter configuration sets a severity level of `*error` for all subcomponents of the PE component. The following table shows the default configuration for each log filter.

| Protocol Engine | Default Log Filter Configuration |
|---------------------------|--|
| PE Manager | pem/*/*error proxy/*/*error |
| Execution Protocol Engine | execpe/*/*error pem/*/*error launchhelper/*/*error |
| X Protocol Engine | xpe/*/*error pem/*/*error |
| Character Protocol Engine | tpe/*/*error pem/*/*error |
| Print Protocol Engine | ppe/*/*error pem/*/*error |
| Channel Protocol Engine | cpe/*/*error pem/*/*error |

For example, to reset an X Protocol Engine log filter use the following command:

```
$ tarantella config edit \  
--tarantella-config-xpeconfig-logfilter "xpe/*/*error" "pem/*/*error"
```

SGD Web Server Logging

SGD web server messages are recorded in the following logs:

- Tomcat JSP technology container logs
- Apache web server logs

Tomcat JSP Technology Container Logs

Log messages for the Tomcat JSP technology container component of the SGD web server are written to the following files in the `/opt/tarantella/webserver/tomcat/tomcat-version/logs` directory on the SGD host:

- `catalina.out` – This log file is rotated when full, or when the Tomcat JSP technology container is restarted, and the contents are appended to `catalina.out.sav`.
- `localhost_log.date.txt` – This is a daily log file, where *date* is the date that messages were recorded.

You can read these log files with a text editor.

The Tomcat JSP technology container log files can be used to diagnose problems with the following:

- Starting and stopping the Tomcat JSP technology container
- Starting the AJP 1.3 Connector
- Loading the SGD webtop web application
- Webtop JSP software exception errors

Logging properties for the Tomcat JSP technology container are configured in the `/opt/tarantella/webserver/tomcat/tomcat-version/conf/logging.properties` file. See the [Tomcat documentation](#) for details of how to configure logging for a Tomcat JSP technology container.

Apache Web Server Logs

Log messages for the Apache web server component of the SGD web server are written to the following files in the `/opt/tarantella/webserver/apache/apache-version/logs` directory on the SGD host:

- `errors_log` – Logs error messages for the Apache web server
- `access_log` – Logs all access requests processed by the Apache web server

You can read these log files with a text editor.

The Apache web server log files can be used to diagnose problems with the following:

- Starting and stopping the Apache web server
- Client requests for SGD webtop pages
- Web server authentication

See the [Apache documentation](#) for more details about these log files.

SGD Client Logging

By default, log messages for the SGD Client are stored to a file, `tcc.txt`, in the following locations on the client device:

- **Microsoft Windows client devices.** A user-specific writeable directory.

On Microsoft Windows XP platforms, for example:

```
C:\Documents and Settings\username\Local Settings\Temp\tcc\
version
```

On Microsoft Windows Vista platforms, for example:

```
C:\Users\username\AppData\Local\Temp\tcc\version
```

The actual location depends on the user's privileges, the operating system, and the version of the Java™ Plugin tool being used.

- **UNIX, Linux, or Mac OS X platform client devices.** The user's home directory:

```
$HOME/.tarantella/tcc/version
```

Users can view the contents of this file with a text editor.

The SGD Client log file can be used to diagnose problems with the following:

- Starting the SGD Client and the SGD Client Helper
- Loading the SGD webtop page
- Client devices, such as CDM, printing, and smart card services
- Connections between the SGD Client and the SGD server

Users configure the logging level for SGD Client messages by changing the Logging Level setting in their client profile. The available logging levels, arranged in increasing level of verbosity, are:

- **No Logging** – Turns off SGD Client logging.
- **Errors only** – Records error messages. This is the default setting.
- **Errors and Warnings only** – Records error messages and warning messages.

- **All** – Records all messages, including error messages, warning messages, and information messages.

Client device information is shown on the Info → Detailed Diagnostics page of the user's webtop.

Administrators can use the Administration Console to view client device information for a user session. Select the user session in the User Session List table and click the View Details button to show more details.

SGD Server Certificate Stores

Each SGD server has two certificate stores, a CA certificate truststore and a client certificate store.

The CA Certificate Truststore

Each SGD server has its own CA certificate truststore. This is the `/opt/tarantella/bin/jre/lib/security/cacerts` file.

The CA certificate truststore contains the CA certificates that the SGD server trusts.

The `/opt/tarantella/etc/data/cacerts.txt` file contains the X.500 Distinguished Names (DNs) and MD5 signatures of all the CA certificates that are in the CA certificates truststore when SGD is first installed. These are the CAs that SGD supports by default. To add support for additional CAs, you can import CA certificates to the truststore.

You might need to import CA certificates in the following circumstances:

- **Active Directory authentication** – If SSL connections to Active Directory are used, and the SSL certificate for an Active Directory server is signed by an unsupported CA, or by an Intermediate CA
See [“SSL Connections to Active Directory” on page 56](#).
- **LDAP authentication** – If SSL connections to LDAP directories are used, and the SSL certificate for an LDAP directory server is signed by an unsupported CA, or by an Intermediate CA
See [“Network Requirements for LDAP Authentication” on page 65](#).

The certificates that must be imported are as follows:

- **Unsupported CA** – Import the CA or root certificate
- **Intermediate CA** – Import the CA certificate chain

If the `tarantella security customca` command is used to install a CA certificate, or CA certificate chain, this command also imports the CA certificates into the CA certificate truststore. It only does this on the SGD server on which the command is run.

To manually import CA certificates, use the `keytool` application. See the [JDK Tools and Utilities](#) documentation for details on how to use the `keytool` application. The `/opt/tarantella/var/tsp/ca.pem` file on the SGD host contains the CA certificate or certificate chain.

If you need to import a CA certificate chain, import each certificate in the chain separately.

The password for the CA certificate truststore is `changeit`.

▼ How to Import CA Certificates or Certificate Chains into the CA Certificate Truststore

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

Repeat the following procedure on each SGD server in the array.

1. Log in as superuser (root) on the SGD host.

2. Import the CA certificate.

To import a CA certificate chain, you must import each certificate in the chain separately.

Use the following command:

```
# /opt/tarantella/bin/jre/bin/keytool -importcert \  
-keystore /opt/tarantella/bin/jre/lib/security/cacerts \  
-storepass changeit -file CA-certificate-path \  
-alias alias
```

Use the `-alias` option to uniquely identify the certificate.

3. Restart the SGD server.

You must restart the SGD server for the CA certificate to become effective.

The Client Certificate Store

Each SGD server has its own client certificate store. This is the `/opt/tarantella/var/info/certs/sslkeystore` file.

The client certificate store contains the client certificates that an SGD server uses to identify itself when connecting to another server.

You create and install server client certificates with the `keytool` application. See the [JDK Tools and Utilities](#) documentation for details on how to use the `keytool` application.

You must provide a password when adding or removing certificates from the client certificate store. The password for the client certificate store is unique to each SGD server and can be found in the `/opt/tarantella/var/info/key` file. Use this password for both the `-storepass` and `-keypass` options.

▼ How to Create a Client Certificate CSR for an SGD Server

1. Log in as superuser (root) on the SGD host.
2. Generate the key pair for the client certificate.

```
# /opt/tarantella/bin/jre/bin/keytool -genkeypair \  
-keyalg rsa \  
-keystore /opt/tarantella/var/info/certs/sslkeystore \  
-storepass "$(cat /opt/tarantella/var/info/key)" \  
-alias alias \  
-keypass "$(cat /opt/tarantella/var/info/key)"
```

Use the `-alias` option to uniquely identify the key pair.

3. Generate a Certificate Signing Request (CSR) for the client certificate.

```
# /opt/tarantella/bin/jre/bin/keytool -certreq \  
-keystore /opt/tarantella/var/info/certs/sslkeystore \  
-storepass "$(cat /opt/tarantella/var/info/key)" \  
-alias alias \  
-keypass "$(cat /opt/tarantella/var/info/key)" \  
-file CSR-path
```

The *alias* must be the same as the alias used when generating the key pair. Aliases are case-insensitive.

▼ How to Install a Client Certificate for an SGD Server

Ensure that no users are logged in to the SGD server and that there are no running application sessions, including suspended application sessions.

Repeat the following procedure on each SGD server in the array.

1. Log in as superuser (root) on the SGD host.
2. Install the client certificate.

```
# /opt/tarantella/bin/jre/bin/keytool -importcert \  
-file certificate-path \  
-keystore /opt/tarantella/var/info/certs/sslkeystore \  
-storepass "$(cat /opt/tarantella/var/info/key)" \  
-alias alias \  
-keypass "$(cat /opt/tarantella/var/info/key)"
```

The *alias* must be the same as the alias used when generating the CSR for the client certificate. Aliases are case-insensitive.

3. Restart the SGD server.

You must restart the SGD server for the client certificate to become effective.

SGD Installations

This section describes the files that are included in an SGD installation. Information on backing up and restoring your SGD installation is also included.

This section includes the following topics:

- [“About Your SGD Installation” on page 411](#)
- [“Backing Up and Restoring an SGD Installation” on page 414](#)

About Your SGD Installation

The standard installation directory for SGD is `/opt/tarantella`.

During SGD installation, you have the option of specifying a different installation directory.

You can find out your installation directory from the command line, as follows:

- **Solaris™ Operating System (Solaris OS) platforms.** Use the command:

```
$ pkgparam `pkginfo 'tta.*' | cut -d' ' -f2` INSTDIR
```

- **Linux system platforms.** Use the command:

```
$ rpm -qi tta | grep Relocations
```

The SGD installation directory contains the following subdirectories:

- bin
- etc
- lib
- var
- webserver

The following sections describe the contents of each of these subdirectories, and what each subdirectory is used for.

See also [“Backing Up and Restoring an SGD Installation”](#) on page 414.

bin Directory

The `bin` directory contains the scripts, binaries, and server-side Java™ technology needed to run SGD.

etc Directory

The `etc` directory contains configuration files that control the behavior of SGD and applications displayed through SGD. It contains the subdirectories listed in the following table.

| Subdirectory | Contents |
|-------------------------------|--|
| <code>etc/data</code> | The following configuration files: <ul style="list-style-type: none">• Character application object configuration files:<ul style="list-style-type: none">– Attribute maps (<code>attrmap.txt</code>)– Color maps (<code>colormap.txt</code>)• Printing configuration files:<ul style="list-style-type: none">– Host name maps (<code>hostnamemap.txt</code>)– Printer driver maps (<code>default.printerinfo.txt</code>)– Printer driver to printer type mappings (<code>printertypes.txt</code>)– Printer to user-friendly name mappings (<code>printernamemap.txt</code>)• RGB color names (<code>rgb.txt</code>)• Timezone configuration files• Supported CA certificates (<code>cacerts.txt</code>) |
| <code>etc/data/keymaps</code> | Keyboard map files. |
| <code>etc/fonts</code> | X Window System fonts and additional fonts installed with SGD. |
| <code>etc/pkg</code> | Information about installed SGD packages, for example version compatibility and dependencies. |
| <code>etc/templates</code> | A complete copy of the standard files that are installed in the <code>etc/data</code> directory and the <code>var/serverresources</code> directory. |

lib Directory

The `lib` directory contains shared libraries used by the SGD server and shared libraries that you might need when installing the SGD Client on certain platforms.

var Directory

The `var` directory contains the files that are used by the web server and the files that the SGD server copies to other members of the array. The `var` directory contains many subdirectories, and the important ones are listed in the following table.

| Subdirectory | Contents |
|---|---|
| <code>var/docroot</code> | The HTML pages used by the SGD web server. |
| <code>var/tsp</code> | Server SSL certificates, keys, and CA certificates. |
| <code>var/ens</code> | The local repository, containing the objects in the organizational hierarchy. |
| <code>var/log</code> | SGD server log files. |
| <code>var/print</code> | The print queue and First In First Out (FIFO). |
| <code>var/serverresources/expect</code> | SGD login scripts. |
| <code>var/spool</code> | Files on their way to the print queue. |

webserver Directory

The `webserver` directory contains the scripts, binaries, and server-side Java technology needed to run the SGD web server, web services, and the webtop. The important subdirectories are listed in the following table.

| Subdirectory | Contents |
|---|--|
| <code>apache</code> | All the files needed to configure and run the SGD web server. |
| <code>tomcat</code> | All the files needed to configure and run the Tomcat JSP technology and Java™ Servlet extension servlet container. |
| <code>tomcat/tomcat-version/webapps/axis</code> | Files needed to run SGD web services. The webtop uses web services. |
| <code>tomcat/tomcat-version/webapps/sgd</code> | Files needed to run the webtop, including the SGD Client. |
| <code>tomcat/tomcat-version/shared/lib</code> | |
| <code>tomcat/tomcat-version/shared/classes</code> | |

Backing Up and Restoring an SGD Installation

This section describes how to back up an SGD installation, so that you can repair SGD in the event that a component or an entire installation becomes damaged.

Before using the procedures on this page, it is helpful if you are familiar with the layout of the SGD installation. See [“About Your SGD Installation” on page 411](#).

This section includes the following topics:

- [“How to Make a Full Backup of an SGD Installation” on page 415](#)
- [“Restoring a Damaged SGD Component” on page 415](#)
- [“How to Do a Full Restore of an SGD Installation” on page 418](#)

▼ How to Make a Full Backup of an SGD Installation

To be able to restore an SGD installation or to be able to repair some individual SGD components, you need a full backup.

While making the backup, *do not* run any command-line tools or use the Administration Console. It is also best if you shut down the SGD server while making the backup. However, if this is not possible, do it when the server is least loaded.

1. **Log on as superuser (root) on the SGD host.**
2. **Back up the SGD log files.**

```
# tarantella archive
```

3. **Back up the entire SGD installation directory on each SGD server in the array.**

See [“About Your SGD Installation” on page 411](#) for details of the SGD installation directory.

SGD also uses the following configuration files, which only need to be backed up if you are using them and you have modified them:

- The `/etc/ttapiprinter.conf` file – This file contains the `lpr` defaults
- The `/etc/sdace.txt` and `/var/ace/data` files – These files contain RSA SecurID settings
- **Web server password files** – If you have created these files for use with the SGD web server, and they are stored outside the SGD installation directory

Restoring a Damaged SGD Component

For the purposes of restoring a damaged installation, SGD can be divided up into the following components:

- Binaries, scripts, and template files
- Login scripts

- Server configuration
- Global configuration
- The local repository
- Automatic log archives
- SGD printing
- The SGD web server, web services, and the webtop

The following sections describe how to back up each of these components.

Binaries, Scripts, and Template Files

The binaries, scripts, and template files are only modified as part of an installation, patch, or custom engineering work. These files do not change very often.

You can restore these files from a backup or another installation, as follows:

- The binaries are in the `/opt/tarantella/bin/bin` directory
- The scripts are in the `/opt/tarantella/bin/scripts` directory
- The template files are in the `/opt/tarantella/etc/templates` directory

Login Scripts

The Login Scripts control the interaction between SGD and the application servers, for example, by logging a user in.

How you recover login scripts depends on whether or not you are using customized login scripts.

If you are not using customized login scripts, you can restore these files from another installation, a backup, or from the `/opt/tarantella/etc/templates` directory.

If you are using customized login scripts, you must only restore these files from a backup.

The login scripts are in the `/opt/tarantella/var/serverresources/expect` directory.

Server Configuration

Server configuration covers all the properties for an SGD server that are not shared with the other SGD servers in the array, such as the server DNS name and server tuning.

As this configuration is unique to a particular SGD host, it must only be restored from a backup taken from that host.

The server-specific configuration is in the
`/opt/tarantella/var/serverconfig/local` directory.

If you are using SGD security services, you must also restore the following:

- `/opt/tarantella/var/tsp`
- `/opt/tarantella/var/info/certs`
- `/opt/tarantella/var/info/key`

Global Configuration

Global configuration covers all the properties that are the same for all the SGD servers in the array, for example the names of the other array members.

To restore the global configuration for an SGD server, you must only restore from a backup of the primary SGD server.

The global configuration is in the
`/opt/tarantella/var/serverconfig/global` directory.

The Local Repository

The local repository, formerly called the Enterprise Naming Scheme (ENS) datastore, is shared across all SGD servers in the array. This is the organizational hierarchy that contains all the information about users, applications, and application servers. This information changes very often.

Restore the local repository from the backup of the primary SGD server.

The local repository is in the `/opt/tarantella/var/ens` directory.

Automatic Log Archives

By default, SGD archives its log files each week at 4 a.m. on Sunday, using a cron job.

If the root user's crontab becomes corrupt, or the archiving does not take place, use the `tarantella setup` command to restore the default setting, or to change the time and day that the archiving takes place.

The log files are archived under the `/opt/tarantella/var/log` directory.

SGD Printing

When you install SGD, it configures an SGD printer queue.

If the printer queue is not present, you can restore it using either of the following methods:

- Use the SGD printer queue installation script, `prtinstall.en.sh`. See [“The SGD Printer Queue Installation Script” on page 230](#).
- Use the `tarantella setup` command.

The printer queue is in the `/opt/tarantella/var/print` directory.

SGD Web Server, Web Services, and the Webtop

The configuration of the SGD web server, SGD web services, and the webtop is unique to a particular SGD host and must only be restored from a backup taken from that host.

The configuration for the SGD web server is in the `/opt/tarantella/webserver/apache/apache-version` directory. You might also have web server password files, which can be stored in other locations.

The configuration for SGD web services is in the `/opt/tarantella/webserver/tomcat/tomcat-version` directory.

The files used for the webtop are in the `/opt/tarantella/webserver/tomcat/tomcat-version/webapps/sgd` directory.

▼ How to Do a Full Restore of an SGD Installation

If you are unable to restore a damaged SGD component or you are unsure about the extent of the damage to your system, you must do a full restore of your SGD installation.

To do a full restore, you must have a full backup. See [“How to Make a Full Backup of an SGD Installation” on page 415](#) for details of how to back up an SGD installation.

Ensure that no users are logged in to the SGD server, and that there are no application sessions, including suspended application sessions, running on the SGD server.

- 1. Log on as superuser (root) on the SGD host.**
- 2. Stop the SGD server.**

3. Uninstall SGD.

```
# tarantella uninstall --purge
```

Note – If this fails, you might have to manually remove the SGD package. Use the `rpm -e tta` command on Linux platforms, and the `pkgrm tta` command on Solaris OS platforms.

4. Delete the SGD installation directory.

```
# rm -rf /opt/tarantella
```

5. Reinstall SGD and any patches, if applicable.

This installs the printer queue, rc scripts and package database.

6. Stop the SGD server.

7. Delete the SGD installation directory.

```
# rm -rf /opt/tarantella
```

8. Reinstate the SGD installation from the backup.

Note – Make sure you restore from the server's backup. Also, check that the DNS name of the host has not changed.

9. Restart the SGD server.

Troubleshooting Arrays and Load Balancing

This section describes some typical problems when using SGD servers, and how to fix them.

The following troubleshooting topics are covered:

- [“Troubleshooting Array Resilience” on page 420](#)
- [“Troubleshooting Clock Synchronization Issues” on page 422](#)
- [“Troubleshooting Advanced Load Management” on page 423](#)

- [“SGD Uses Too Much Network Bandwidth” on page 427](#)
- [“Users Cannot Connect to an SGD Server When It Is In Firewall Traversal Mode” on page 428](#)
- [“Users Cannot Relocate Their Sessions” on page 429](#)

Troubleshooting Array Resilience

To help you to diagnose and fix problems when using array resilience, you can do the following:

- Show status information for the SGD array
- Enable array resilience logging

Showing Status Information For an SGD Array

You use the `tarantella status` command on an SGD server to show status information for the server.

This section includes some examples of using `tarantella status` to show status information for an SGD array when the primary server in the array goes down. [“Primary Server Goes Down” on page 341](#) includes a detailed description of this array resilience scenario.

The original network configuration used for the examples is a three-node array of SGD servers in the domain `example.com`, as follows:

- **Primary server** – `boston`
- **Secondary servers** – `newyork`, `detroit`

When the primary server `boston` goes down, running `tarantella status` on `newyork` indicates that there is a connection problem with the SGD array, as follows:

```
$ tarantella status
Array members (3):
- newyork.example.com (secondary): Accepting standard connections.
- boston.example.com (primary): NOT ACCEPTING CONNECTIONS.
- detroit.example.com (secondary): Accepting standard connections.
...
```

If the SGD servers in the array do not agree on the array membership, `tarantella status` shows the array configuration as seen by every SGD server in the array. For example, running `tarantella status` on `newyork` during the failover stage might show the following information:

```
$ tarantella status
Inconsistent array: the servers report different array membership.
...
boston.example.com reports an error:
- Host is unavailable

newyork.example.com reports 3 members as:
- newyork.example.com
- boston.example.com
- detroit.example.com

detroit.example.com reports 1 member as:
- detroit.example.com
```

The `tarantella status` command indicates if the array is in a repaired state. For example, running `tarantella status` from `detroit` after the failover stage has completed might show the following information:

```
$ tarantella status
Array members (2):
- newyork.example.com (primary)
- detroit.example.com (secondary)
...
This node is in a repaired array. Any alterations to array state will prevent
recovery of the original array.
Use the tarantella status --originalstate command to see the original array
state.
```

You use the `--originalstate` option to list the members of the array before it was repaired. For example, using the `--originalstate` option on any server in the array shows the original array members, as follows:

```
$ tarantella status --originalstate
Original array members (3):
- boston.example.com (primary)
- newyork.example.com (secondary)
- detroit.example.com (secondary)
...
```

After the recovery stage, you can use the `tarantella status` command to verify that the original array formation has been recreated. For example, running `tarantella status` on `newyork` might display the following information:

```
$ tarantella status
Array members (3):
- newyork.example.com (secondary): Accepting standard connections.
- boston.example.com (primary): Accepting standard connections.
- detroit.example.com (secondary): Accepting standard connections.
...
```

Enabling Array Resilience Logging

To enable logging for array resilience, add the following log filters in the Log Filter field on the Global Settings → Monitoring tab in the Administration Console:

```
server/failover/*:failover%%PID%%.log
server/failover/*:failover%%PID%%.jsl
```

See [“Using Log Filters to Troubleshoot Problems With an SGD Server”](#) on page 391 for more information on configuring and using SGD log filters.

Troubleshooting Clock Synchronization Issues

Problems can arise if the clocks on the SGD servers in an array are not in synchronization. If possible, use NTP software or the `rdate` command to ensure the clocks on all SGD hosts are synchronized.

You run the `tarantella status` command on the primary SGD server to show any clock synchronization issues for the array. The following example indicates that the clock on the secondary server `newyork.example.com` is out of synchronization.

```
$ tarantella status
Array members (3):
- boston.example.com (primary): Accepting standard connections.
- newyork.example.com (secondary): Accepting standard connections.
- detroit.example.com (secondary): Accepting standard connections.

WARNING: The clocks on the array nodes are not synchronized.
The following array members disagree with the primary:
- newyork.example.com
```

If clocks are out of synchronization, a warning message is also displayed on the Secure Global Desktop Servers tab of the Administration Console.

You use the `--byserver` option of `tarantella status` to display the clock setting on each SGD server in the array, as follows:

```
$ tarantella status --byserver

boston.example.com:
- Array member (primary): Accepting standard connections.
...
- Current time reported: Wed Apr 28 09:36:16 BST 2010

newyork.example.com:
- Array member (secondary): Accepting standard connections.
...
- Current time reported: Wed Apr 28 09:38:02 BST 2010

detroit.example.com:
- Array member (secondary): Accepting standard connections.
...
- Current time reported: Wed Apr 28 09:36:16 BST 2010

WARNING: The clocks on the array nodes are not synchronized.
```

Troubleshooting Advanced Load Management

If you experience problems with the Least CPU Usage and Most Free Memory methods of application load balancing, you can get information from the following places to help you understand what is happening:

- SGD server log files

Add the following filters to the Log Filters field on the Global Settings → Monitoring tab in the Administration Console:

```
server/tier3loadbalancing/*:t3loadbal%%PID%%.log
server/tier3loadbalancing/*:t3loadbal%%PID%%.jsl
```

This provides detailed information about the decision to run an application and the data being sent by the application server.

See [“Using Log Filters to Troubleshoot Problems With an SGD Server” on page 391](#) for more information on configuring and using SGD log filters.

- SGD Enhancement Module logs

For UNIX or Linux platform application servers, these are in the `/opt/tta_tem/var/log/tier3loadprobePID_error.log` file.

For Windows application servers, this information is displayed in the Event Viewer.

- Load balancing service connection Common Gateway Interface (CGI) program
Go to the `http://applicationserver:3579?get&ttaalbinfo` URL.

You can use this information to troubleshoot the following common problems:

- [“The Load Balancing Service Is Not Working” on page 424](#)
- [“SGD Ignores an Application Server Load Balancing Properties File” on page 425](#)
- [“One of the Application Servers Is Never Picked” on page 425](#)
- [“One of the Application Servers Is Always Picked” on page 426](#)
- [“Two Identical Application Servers, But One Runs More Applications Than the Other” on page 427](#)
- [“The SGD Server Log File Shows an Update Received for an Unknown ID” on page 427](#)

The Load Balancing Service Is Not Working

If you think the load balancing service is not working, check the following.

Is the SGD Enhancement Module installed and running?

On Microsoft Windows applications servers, use Control Panel → Administrative Tools → Services to check whether the Tarantella Load Balancing Service is listed and is started.

On UNIX and Linux platform application servers, run the following command as superuser (root) to check that load balancing processes are running:

```
# /opt/tta_tem/bin/tem status
```

Is the primary SGD server running?

The load balancing service on the application server sends load information to the primary SGD server. If the primary is not available, SGD uses Fewest application sessions as the method for load balancing application servers.

Is your firewall blocking the load balancing service?

For the load balancing service to work, the firewall must allow the following connections:

- A TCP connection on port 3579 between the SGD server and the application server.

- A UDP connection on port 3579 between the application server and the SGD server.

Note – These connections do not need to be authenticated.

What do the log files show?

Check the log files for further information, see [“Troubleshooting Advanced Load Management” on page 423](#) for details.

SGD Ignores an Application Server Load Balancing Properties File

After creating a load balancing properties file for an application server, you must do a warm restart of the primary SGD server. Run the following command as superuser (root):

```
# tarantella restart sgd --warm
```

Ensure that no users are logged in to the SGD server, and that there are no application sessions, including suspended application sessions, running on the SGD server.

One of the Application Servers Is Never Picked

If one of the application servers is never picked to run applications, check the following.

Is the load balancing service running on the application server?

See [“The Load Balancing Service Is Not Working” on page 424](#).

Is the application server available to run applications?

Check the application server object in the Administration Console. Ensure the Application Start check box is selected on the General tab for the application server object.

Check that the application server is up.

What do the log files show?

Check the log files for further information, see [“Troubleshooting Advanced Load Management” on page 423](#) for details.

One of the Application Servers Is Always Picked

If one application server is always picked to run applications regardless of its load, check the following.

Is more than one application server configured to run the application?

Check the Hosting Application Servers tab for the application object.

Are the other application servers available to run applications?

Check the application server objects in the Administration Console. Ensure the Application Start check box is selected on the General tab

Check that all the application servers are up.

Is the correct load balancing method selected?

In the Administration Console, check that either Most Free Memory or Least CPU Usage is selected as the load balancing method on the Performance tab for the application object, or on the Global Settings → Performance tab.

Are you using server affinity?

Server affinity means that, if possible, SGD starts an application on the same application server as the last application started by the user. Server affinity is on by default, see [“Server Affinity” on page 366](#).

Is the load balancing service running on the application server?

See [“The Load Balancing Service Is Not Working” on page 424](#).

What do the log files show?

Check the log files for further information, see [“Troubleshooting Advanced Load Management” on page 423](#) for details.

Two Identical Application Servers, But One Runs More Applications Than the Other

Check that the server weighting value for the servers are the same. See [“Application Server’s Relative Power” on page 373](#).

The SGD Server Log File Shows an Update Received for an Unknown ID

The SGD server log file might show an information message containing the following text:

```
Got an update for unknown id from machine applicationserver
```

This message can be ignored. It occurs only when the primary SGD server is restarted.

SGD Uses Too Much Network Bandwidth

If SGD is using a lot of network bandwidth, set the Bandwidth Limit attribute for a user profile to reduce the maximum allowable bandwidth the user can use.

Note – Reducing the available bandwidth might have implications for application usability.

In the Administration Console, go to the User Profiles tab and select the user profile object you want to configure. Go to the Performance tab and select a value from the Bandwidth Limit list.

Alternatively, use the following command:

```
$ tarantella object edit --name obj --bandwidth bandwidth
```

The following are the available bandwidths:

| Administration Console | Command Line |
|------------------------|--------------|
| 2400 bps | 2400 |
| 4800 bps | 4800 |
| 9600 bps | 9600 |

| Administration Console | Command Line |
|------------------------|--------------|
| 14.4 Kbps | 14400 |
| 19.2 Kbps | 19200 |
| 28.8 Kbps | 28800 |
| 33.6 Kbps | 33600 |
| 38.8 Kbps | 38800 |
| 57.6 Kbps | 57600 |
| 64 Kbps | 64000 |
| 128 Kbps | 128000 |
| 256 Kbps | 256000 |
| 512 Kbps | 512000 |
| 768 Kbps | 768000 |
| 1 Mbps | 1000000 |
| 1.5 Mbps | 1500000 |
| 10 Mbps | 10000000 |
| None | 0 |

Note – None is the default. This means there is no limit on bandwidth usage.

Users Cannot Connect to an SGD Server When It Is In Firewall Traversal Mode

If users cannot connect to an SGD server when it is in firewall traversal mode, this is usually caused by starting the SGD server *before* the SGD web server.

In firewall traversal mode, an SGD server listens on port 443 and forwards any web connections to the SGD web server, which is configured to listen on `localhost` port 443 (`127.0.0.1:443`).

If an SGD server is started before the SGD web server, the SGD server binds to all the available interfaces and this means that the SGD server forwards any web connections to itself in an infinite loop.

One solution is to always start the SGD web server before the SGD server. If you use the `tarantella start` command, the SGD server and web server are always started in the correct order.

Another solution is to configure SGD so that it never binds to the `localhost` interface. To do this, use the following command:

```
$ tarantella config edit \  
--tarantella-config-server-bindaddresses-external "!127.0.0.1"
```

Note – On some shells you cannot use straight quotation marks, `"!127.0.0.1"`, as the `!127` might be substituted. Use single straight quotation marks instead, `'!127.0.0.1'`.

You can also use this command to specify exactly which interfaces you do want SGD to bind to. You do this by typing a comma-separated list of DNS names or IP addresses.

See [“Firewall Traversal” on page 24](#) for more details about running SGD in firewall traversal mode.

Users Cannot Relocate Their Sessions

When a user logs in to an SGD server without logging out of another, normally the user’s session is relocated to the new server. This is sometimes called session moving, or session grabbing.

If the clocks on all SGD servers in the array are not synchronized, user sessions might not relocate successfully.

SGD uses the time stamps on user sessions to determine which is newer. The newer user session is considered to be current. If clocks are not synchronized, the time stamps might give misleading information.

Because time synchronization is important, use Network Time Protocol (NTP) software to synchronize clocks. Alternatively, use the `rdate` command.

See also [“User Sessions and Application Sessions” on page 387](#) for more information about user sessions in SGD.

Global Settings and Caches

Use the Global Settings tabs to configure settings that apply to Oracle Secure Global Desktop (SGD) as a whole. Changes made in the Global Settings tabs affect all SGD servers in the array.

Use the Caches tab to view and manage entries in the password cache and the token cache.

This chapter includes the following topics:

- [“Secure Global Desktop Authentication Tab” on page 432](#)
- [“Service Objects Tab” on page 446](#)
- [“Application Authentication Tab” on page 451](#)
- [“Communication Tab” on page 457](#)
- [“Performance Tab” on page 462](#)
- [“Client Device Tab” on page 465](#)
- [“Printing Tab” on page 474](#)
- [“Security Tab” on page 478](#)
- [“Monitoring Tab” on page 481](#)
- [“Resilience Tab” on page 483](#)
- [“Caches Tab” on page 488](#)
- [“Passwords Tab” on page 488](#)
- [“Tokens Tab” on page 490](#)

Secure Global Desktop Authentication Tab

Use the settings on the Secure Global Desktop Authentication tab to control how users log in to SGD. The settings apply to all SGD servers in the array. Changes to the settings take effect immediately.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

User authentication can be performed by an external authentication mechanism (*third-party authentication*), or SGD can perform the authentication using a specified repository (*system authentication*).

The Secure Global Desktop Authentication tab contains the following sections:

- **Tokens and Cache.** This section contains the following attributes:
 - [Token Generation](#)
 - [Password Cache](#)
- **Secure Global Desktop Effective Sequence.** This section displays a summary of the current SGD authentication settings. If you click the Change User Authentication button, the Authentication Wizard starts. The Wizard enables you to configure SGD authentication. See [The Authentication Wizard](#).

The Authentication Wizard

The Authentication Wizard guides you through the process of setting up authentication for SGD users. The number of steps shown in the Authentication Wizard depend on the choices you make as you work through the Wizard.

The available steps in the Authentication Wizard are as follows:

- **Overview.** Includes background information about how users authenticate to SGD.
- **Third-Party/System Authentication.** Select whether you want to use third-party authentication, system authentication or both.

This step contains the following attributes:

- [Third-Party Authentication](#)
- [System Authentication](#)

- **Third-Party Authentication – User Identity and Profile.** For third-party authentication only. Choose search methods to use for finding the user identity and user profile of the authenticated user.

This step contains the following attributes:

- [Search Local Repository](#)
 - [Search LDAP Repository](#)
 - [Use Default Third-Party Identity](#)
 - [Use Default LDAP Profile](#)
 - [Use Closest Matching LDAP Profile](#)
- **System Authentication – Repositories.** For system authentication only. Select one or more check boxes to enable repositories that SGD uses for locating user information. The repositories are listed in the order in which they are tried. If one repository authenticates the user, no more repositories are tried.

This step contains the following attributes:

- [LDAP/Active Directory](#)
 - [Unix](#)
 - [Authentication Token](#)
 - [Windows Domain Controller](#)
 - [SecurID](#)
 - [Anonymous](#)
- **Unix Authentication – User Profile.** For system authentication only. This screen is shown if UNIX authentication is selected. Select one or more check boxes to specify how to find the user profile for the authenticated UNIX® system user. The authentication methods are listed in the order in which they are tried. If one method finds a matching user profile, no more search methods are tried.

This step contains the following attributes:

- [Search Unix User ID in Local Repository](#)
 - [Search Unix Group ID in Local Repository](#)
 - [Use Default User Profile](#)
- **Windows Domain Authentication – Domain Controller.** For system authentication only. This screen is shown if the Windows Domain Controller system authentication repository is selected. Here, you specify the name of the domain controller.

This step contains the [Windows Domain](#) attribute.

- **LDAP Repository Details.** For third-party or system authentication. This screen is shown if an LDAP or Active Directory system authentication repository is selected, or if the Search LDAP Repository option is selected for third-party authentication. Here, you specify details of the LDAP repository to use.

This step contains the following attributes:

- [Active Directory](#)
- [LDAP](#)
- [URLs](#)
- [User Name and Password](#)
- [Connection Security](#)
- [Active Directory Base Domain](#)
- [Active Directory Default Domain](#)

The LDAP Repository Details step enables you to create and manage the service object called `generated`. If more than one service object is configured, you use the Service Object tab to configure these details, see [“Service Objects Tab” on page 446](#).

- **Review Selections.** Shows a summary of the choices you have made using the Wizard. You can review your authentication settings before confirming the changes.

Token Generation

Usage: Select or deselect the check box.

Description

Whether to create authentication tokens for users so they can log in automatically to SGD.

To ensure that an authentication token cannot be intercepted and used by a third party, use secure Hypertext Transfer Protocol over Secure Socket Layer (HTTPS) web servers and enable SGD security services.

Command Line

Command option: `--login-autotoken 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables generation of authentication tokens for users.

```
--login-autotoken 0
```

Password Cache

Usage: Select or deselect the check box.

Description

Whether to save the user name and password that the user types to log in to SGD in the password cache.

If you are using SecurID authentication, do not save the user name and password, as SecurID passwords cannot be reused.

SGD cannot store the user names and passwords of users authenticated with third-party authentication.

Command Line

Command option: `--launch-savetapassword 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example saves user log in details in the password cache.

```
--launch-savetapassword 1
```

Third-Party Authentication

Usage: Select or deselect the check box.

Description

Select the check box to enable third-party authentication.

This attribute enables you to give access to SGD to users who have been authenticated by a third-party mechanism, such as web server authentication.

Command Line

Command option: `--login-thirdparty 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables third-party authentication.

```
--login-thirdparty 0
```

System Authentication

Usage: Select or deselect the check box.

Description

Specifies that user authentication is done by the SGD server. Selecting this option enables the Wizard screens for system authentication settings.

Command Line

There is no command line equivalent for this attribute.

Search Local Repository

Usage: Select or deselect the check box.

Description

This attribute specifies a search method used by SGD to determine the identity and user profile of a user who has been authenticated by a third-party authentication mechanism.

This search method searches for the user identity in the local repository and then uses the matching user profile.

If additional search methods are selected, the search methods are used in the order shown. However, third-party authentication does not support ambiguous users and so the first match found is used.

If the searches do not produce a match, the standard login page is displayed and the user must log in to SGD in the normal way.

Command Line

Command option: `--login-thirdparty-ens 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, searching the local repository for a matching user profile is disabled.

```
--login-thirdparty-ens 0
```

Search LDAP Repository

Usage: Select or deselect the check box.

Description

Specifies that the LDAP repository is searched to find the user identity for a user who has been authenticated by a third-party authentication mechanism.

The search method used is defined by the [Use Default LDAP Profile](#) or [Use Closest Matching LDAP Profile](#) attribute.

Command Line

There is no command line equivalent for this attribute.

Use Default Third-Party Identity

Usage: Select or deselect the check box.

Description

This attribute specifies a search method used by SGD to determine the identity and user profile of a user who has been authenticated by a third-party authentication mechanism.

This search method does not perform a search. The user identity is the third-party user name. The third-party user profile, `System Objects/Third Party Profile`, is used.

If additional search methods are selected, the search methods are used in the order shown. However, third-party authentication does not support ambiguous users and so the first match found is used.

If the searches do not produce a match, the standard login page is displayed and the user must log in to SGD in the normal way.

Command Line

Command option: `--login-thirdparty-nonens 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, using the default user profile is disabled.

```
--login-thirdparty-nonens 0
```

Use Default LDAP Profile

Usage: Select the option.

Description

This attribute specifies a search method used by SGD to determine the identity and user profile of a user who has been authenticated by a third-party authentication mechanism.

This search method searches for the user identity in an LDAP repository and then uses the default LDAP user profile, `System Objects/LDAP Profile`.

If additional search methods are selected, the search methods are used in the order shown. However, third-party authentication does not support ambiguous users and so the first match found is used.

If the searches do not produce a match, the standard login page is displayed and the user must log in to SGD in the normal way.

Command Line

Command option: `--login-ldap-thirdparty-profile 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, searching LDAP and using the default LDAP profile is disabled.

```
--login-ldap-thirdparty-profile 0
```

Use Closest Matching LDAP Profile

Usage: Select the option.

Description

This attribute specifies a search method used by SGD to determine the identity and user profile of a user who has been authenticated by a third-party authentication mechanism.

This search method searches for the user identity in an LDAP repository and then uses the closest matching user profile in the local repository, allowing for differences between the LDAP and SGD naming systems.

SGD searches for the following until a match is found:

- A user profile with the same name as the LDAP person object.
For example, if the LDAP person object is `cn=Emma Rald,cn=Sales,dc=example,dc=com`, SGD searches the local repository for `dc=com/dc=example/cn=Sales/cn=Emma Rald`.
- A user profile in the same organizational unit as the LDAP person object but with the name `cn=LDAP Profile`.
For example, `dc=com/dc=example/cn=Sales/cn=LDAP Profile`.
- A user profile in any parent organizational unit with the name `cn=LDAP Profile`.
For example, `dc=com/dc=example/cn=LDAP Profile`.
- If there is no match, the profile object `System Objects/LDAP Profile` is used for the user profile.

If additional search methods are selected, the search methods are used in the order shown. However, third-party authentication does not support ambiguous users and so the first match found is used.

If the searches do not produce a match, the standard login page is displayed and the user must log in to SGD in the normal way.

Command Line

Command option: `--login-ldap-thirdparty-ens 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, searching LDAP and using the closest matching LDAP profile is disabled.

```
--login-ldap-thirdparty-ens 0
```

LDAP/Active Directory

Usage: Select or deselect the check box.

Description

Specifies that an LDAP directory server or Active Directory server is used for authentication.

Selecting this option enables the Wizard screen where you can type in LDAP directory server or Active Directory server details.

Command Line

There is no command line equivalent for this attribute.

Unix

Usage: Select or deselect the check box.

Description

Enables UNIX authentication.

Selecting this option enables the Wizard screen where you can configure UNIX authentication settings.

Command Line

There is no command line equivalent for this attribute.

Authentication Token

Usage: Select or deselect the check box.

Description

Enables authentication using an authentication token.

Authentication using an authentication token can only be used when the SGD Client is operating in Integrated mode.

Command Line

Command option: `--login-atla 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, authentication using an authentication token is disabled.

```
--login-atla 0
```

Windows Domain Controller

Usage: Select or deselect the check box.

Description

Enables authentication against a Windows domain controller.

Command Line

Command option: `--login-nt 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, Windows Domain Controller authentication is disabled.

```
--login-nt 0
```

SecurID

Usage: Select or deselect the check box.

Description

Enables users with RSA SecurID tokens to log in to SGD.

Command Line

Command option: --login-securid 1 | 0

Usage: Specify 1 (true) or 0 (false).

In the following example, SecurID authentication is disabled.

```
--login-securid 0
```

Anonymous

Usage: Select or deselect the check box.

Description

Enables users to log in to SGD without supplying a user name and password.

Command Line

Command option: --login-anon 1 | 0

Usage: Specify 1 (true) or 0 (false).

In the following example, anonymous user authentication is disabled.

```
--login-anon 0
```

Search Unix User ID in Local Repository

Usage: Select or deselect the check box.

Description

Specifies a search method used to find the user profile for an authenticated UNIX system user. Select this attribute to search for the user identity in the local repository and use the matching user profile.

Command Line

Command option: `--login-ens 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, searching for the UNIX User ID in the local repository is enabled.

```
--login-ens 1
```

Search Unix Group ID in Local Repository

Usage: Select or deselect the check box.

Description

Specifies a search method used to find the user profile for an authenticated UNIX system user. Select this attribute to use the UNIX user identity and search for a user profile in the local repository that matches the user's UNIX Group ID.

Command Line

Command option: `--login-unix-group 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, searching for the UNIX Group ID in the local repository is enabled.

```
--login-unix-group 1
```

Use Default User Profile

Usage: Select or deselect the check box.

Description

Specifies a search method used to find the user profile for an authenticated UNIX system user. Select this attribute to use the default UNIX user profile, *System Objects/UNIX User Profile*, for the authenticated user.

Command Line

Command option: `--login-unix-user 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, using the default UNIX user profile (*System Objects/UNIX User Profile*) is enabled.

```
--login-unix-user 1
```

Windows Domain

Usage: Type the Windows domain name in the field.

Description

The name of the domain controller used for Windows domain authentication.

Command Line

Command option: `--login-nt-domain dom`

Usage: Replace *dom* with the name of the Windows domain controller used to authenticate users.

In the following example, users are authenticated with the Windows domain controller `sales.indigo-insurance.com`.

```
--login-nt-domain sales.indigo-insurance.com
```

Active Directory

Usage: Select the option.

Description

Enables Active Directory authentication.

Command Line

Command option: `--login-ad 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, Active Directory authentication is enabled.

```
--login-ad 1
```

LDAP

Usage: Select the LDAP option.

Description

Enables LDAP authentication.

Command Line

Command option: `--login-ldap 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, LDAP authentication is enabled.

```
--login-ldap 1
```

Service Objects Tab

The Service Objects tab is where you can view, create, edit, and manage service objects. A service object is a group of configuration settings used for the following SGD authentication mechanisms:

- Active Directory authentication, see [“Active Directory Authentication” on page 52](#)
- LDAP authentication, see [“LDAP Authentication” on page 63](#)
- Third-party authentication using the LDAP repository search, see [“Third-Party Authentication” on page 72](#)

Use the buttons in the Service Objects List table to manage service objects for the SGD array.

Use the Repository Type option to enable either [LDAP](#) or [Active Directory](#) authentication. The Repository Type option is only available if both LDAP and Active Directory service objects have been created.

From the command line, use the `tarantella service` commands to create, delete, edit, and list service objects. See [“The `tarantella service` Command” on page 816](#).

For more information about service objects, see [“Using Service Objects” on page 90](#).

The Service Objects List Table

The Service Objects List table displays the service objects configured for the SGD array.

When you enable LDAP or Active Directory authentication using the Secure Global Desktop Authentication Wizard, a service object called `generated` is created automatically and the Service Objects List table is shown.

The Service Objects List table includes the following information for each service object:

- **Position.** Position of the service object in the table. The highest position is 1. SGD uses the enabled service objects in the order shown.
- **Name.** Name of the service object.

- **Enabled/Disabled.** Whether the service object is enabled or disabled.
- **Type.** Service object type, either LDAP or Active Directory.
- **URL.** URL of the LDAP server or Active Directory forest. Where multiple LDAP servers have been specified, multiple URLs are shown.

The New button is used to create a new service object. The new service object is added at the end of the Service Objects List table in last position.

The Edit button is used to edit the selected service object.

The Delete button removes the selected service object.

The Duplicate button makes a copy of the selected service object.

The Enable and Disable buttons switches the enabled state of the selected service object.

The Move Up and Move Down buttons are used to change the position of the selected service object in the table.

You update the Service Objects List table by clicking the Reload button.

When you create, duplicate, or edit a service object, a new window is displayed that enables you to configure the service object. In this window, you can configure only the following commonly-used settings for service objects:

- [Name](#)
- [Type](#)
- [Enabled](#)
- [URLs](#)
- [User Name and Password](#)
- [Connection Security](#)
- [Active Directory Base Domain](#)
- [Active Directory Default Domain](#)

There are also some advanced service object settings that can be configured only from the command line with the `tarantella service new` or the `tarantella service edit` commands, see [“Using Service Objects” on page 90](#) for more details.

Name

Usage: Type the name of the service object in the field.

The name of the service object.

Once you have created a service object, you cannot rename it. Use the Duplicate button in the Service Objects List table to create a copy of the service object with a different name.

The name can only contain lowercase characters, digits, or the characters “_” and “-”.

Type

Usage: Select either the LDAP or Active Directory option.

The Type setting controls which SGD authentication mechanism can use the service object.

Select the LDAP option even if you are using a Microsoft Active Directory server for LDAP authentication.

Active Directory service objects are used only for Active Directory authentication.

Once you have created a service object, you cannot change the type.

Enabled

Usage: Select or deselect the check box.

Whether to enable the service object. A service object must be enabled before SGD can use it.

URLs

Usage: Type one or more uniform resource locators (URLs) in the field. Separate each URL with a semicolon.

For *LDAP service objects*, type one or more URLs of LDAP directories. The URLs are used in the order they are listed. If the first LDAP directory server listed is unavailable, SGD tries the next one in the list. Alternatively, you can create separate service objects for each URL. SGD uses each service object in their position order. Each LDAP URL has the form `ldap://server:port/searchroot`. Each of these options is defined as follows:

- **Server.** The Domain Name System (DNS) name of the LDAP directory server.
- **Port.** The Transmission Control Protocol (TCP) port that the LDAP directory server listens on for connections. You can omit this, and the preceding “:”, to use the default port.

- **Searchroot.** The distinguished name (DN) to use as the search base, for example, `dc=example,dc=com`. This specifies the part of the LDAP directory used to search for the user identity.

Use an `ldaps://` URL if your LDAP directory server uses Secure Sockets Layer (SSL) connections. Extra configuration might be required for SSL connections, see [“Network Requirements for LDAP Authentication” on page 65](#).

The URLs configured for an LDAP service object must all be of the same type, either `ldap://` or `ldaps://`. You cannot use a mixture of `ldap://` and `ldaps://` URLs.

For *Active Directory service objects*, type a the URL of an Active Directory forest. For example, `ad://example.com`. The URL *must* start `ad://`. Only type *one* URL.

Use the Test button to test the connection to the URLs.

User Name and Password

Usage: Type the user name and password in the fields.

The user name and password of a user that has privileges to search the directory server.

For security reasons, the password is not displayed, even if it has been previously set.

For *LDAP service objects*, type the DN of the user, for example `cn=sgd-user,cn=Users,dc=example,dc=com`. This is the administrator bind DN, see [“LDAP Bind DN and Password Change” on page 66](#) for more details. As you can only enter one user name and password, this user must be able to search all LDAP directory servers listed in the URL field. If you need to use different user names and password, create separate service objects. If the directory server supports anonymous binds, you can omit the user name and password. To use anonymous binds, you must be able to perform LDAP queries for user data.

For *Active Directory service objects*, the user name has the form `user@example.com`. If you omit the domain name from the user name. SGD uses the information in the URL, Base Domain, and Default Domain fields to obtain a domain. The user must have privileges to search Active Directory for user information.

To configure the user name and password for the directory server on the command line, use the `tarantella passcache` command. See [“The tarantella passcache Command” on page 762](#) for more details.

Connection Security

Usage: Select the required option. If the SSL option is selected, an option for using client certificates is enabled.

The mechanism used to secure the connection to an Active Directory server.

- **To use only the Kerberos protocol for secure connections** – Select the Kerberos option for Connection Security, and type a user name and password in the User Name and Password fields. This option is selected by default.
- **To use Kerberos and SSL for secure connections** – Select the SSL option for Connection Security, and type a user name and password in the User Name and Password fields.
- **To use Kerberos, SSL, and client certificates for secure connections** – Select the SSL option for Connection Security, and select the Use Certificates check box.

See [“SSL Connections to Active Directory” on page 56](#) for details of the additional configuration required to use SSL connections.

Active Directory Base Domain

Usage: Type a domain name in the field.

The domain that SGD uses for Active Directory authentication, if users only supply a partial domain when they log in.

For example, if the base domain is set to `example.com` and a user logs in with the user name `rouge@west`, SGD authenticates the user as `rouge@west.example.com`.

Active Directory Default Domain

Usage: Type a domain name in the field.

The domain that SGD uses for Active Directory authentication, if users do not supply a domain when they log in.

For example, if the default domain is set to `east.example.com` and a user logs in with the user name `rouge`, SGD authenticates the user as `rouge@east.example.com`.

Application Authentication Tab

Settings on the Application Authentication tab control the user experience when starting applications.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Changes to these attributes take effect immediately.

This tab contains the following sections:

- Authentication

This section contains the following attributes:

- Password Cache Usage
- Action When Password Expired
- Smart Card Authentication

- Authentication Dialog

This section contains the following attributes:

- Dialog Display
- “Save Password” Box
- “Always Use Smart Card” Box

- Launch Dialog

This section contains the following attributes:

- Display Delay
- “Launch Details” Pane

Password Cache Usage

Usage: Select or deselect the check box.

Description

Whether to try the password the user typed for the SGD server, if it is stored in the password cache, as the password for the application server.

SGD server passwords might be stored in the cache if some applications are configured to run on the SGD host, or if [Password Cache](#) is selected.

This attribute can be overridden by an application server object's [Password Cache Usage](#) attribute.

Command Line

Command option: `--launch-trycachedpassword 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example uses the SGD password stored in the password cache when authenticating to an application server.

`--launch-trycachedpassword 1`

Action When Password Expired

Usage: Select an option.

Description

The action to take if the user's password has expired on the application server.

The command line options and their Administration Console equivalents are shown in the following table.

| Administration Console | Command Line | Description |
|------------------------|---------------------|---|
| Authentication Dialog | <code>dialog</code> | Show an SGD authentication dialog. |
| Aged Password Handler | <code>manual</code> | Show a terminal window, where the user can change their password. |
| Launch Failure | <code>none</code> | Take no further action. Treat as a startup failure. |

For Windows applications, the Terminal Server handles the authentication process. No information is returned to SGD indicating whether authentication succeeds or fails. This means that once SGD has cached a user name and password for the Windows application server, SGD never displays the authentication dialog again unless the user holds down the Shift key when they click an application's link, or an Administrator deletes the user's entry from the password cache.

Command Line

Command option: `--launch-expiredpassword manual | dialog | none`

Usage: Specify an option.

In the following example, the user can change their password using a terminal window.

```
--launch-expiredpassword manual
```

Smart Card Authentication

Usage: Select or deselect the check box.

Description

Enable users to log in to a Microsoft Windows application server with a smart card.

Command Line

Command option: `--launch-allowsmartcard 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables users to log in using a smart card.

```
--launch-allowsmartcard 1
```

Dialog Display

Usage: Select or deselect the check boxes.

Description

Controls when the application server's authentication dialog is displayed. The check boxes are inter-related, enabling you to select from three possible options.

The command line options and their Administration Console equivalents are shown in the following table.

| Administration Console | Command Line | Description |
|---|--------------|---|
| On Shift-Click (selected) On Password Problem (selected) | user | Show the authentication dialog if the user holds down the Shift key when they click an application's link, or if there is a password problem. |
| On Shift-Click (deselected) On Password Problem (selected) | system | Only show the authentication dialog when there is a password problem. |
| On Shift-Click (deselected) On Password Problem (deselected) | none | Never show the authentication dialog. |

For Windows applications, it is the Terminal Server handles the authentication process. No information is returned to SGD indicating whether authentication succeeds or fails. This means that once SGD has cached a user name and password for the Windows application server, SGD never displays the authentication dialog again unless the user holds down the Shift key when they click an application's link, or an Administrator deletes the user's entry from the password cache.

Command Line

Command option: --launch-showauthdialog user | system | none

Usage: Specify an option.

In the following example, the application server's authentication dialog is shown if you hold down the Shift key and click a link to start an application, or if there is a problem with the password.

```
--launch-showauthdialog user
```

"Save Password" Box

Usage: Select or deselect the check boxes.

Description

Two attributes that control the initial state of the Save Password check box in the application server authentication dialog and whether users can change it.

If users cannot change the setting, the Initially Checked attribute determines whether users can save passwords in the application server password cache.

Command Line

Command option: `--launch-savepassword-initial` checked | unchecked

Command option: `--launch-savepassword-state` enabled | disabled

Usage: Specify a valid option.

In the following example, the initial state of the Save Password check box is *selected*. Users can change this setting.

```
--launch-savepassword-initial checked
--launch-savepassword-state enabled
```

“Always Use Smart Card” Box

Usage: Select or deselect the check boxes.

Description

Two attributes that control the initial state of the Always Use Smart Card check box in the application server authentication dialog box and whether users can change it.

If users cannot change the setting, the Initially Checked attribute determines whether the user’s decision to always use smart card authentication is cached.

Command Line

Command option: `--launch-alwayssmartcard-initial` checked|unchecked

Command option: `--launch-alwayssmartcard-state` enabled|disabled

Usage: Specify a valid option.

In the following example, the initial state of the Always Use Smart Card check box is *selected*. Users can change to this setting.

```
--launch-alwayssmartcard-initial checked
--launch-alwayssmartcard-state enabled
```

Display Delay

Usage: Enter a time period, measured in seconds, in the field.

Description

The delay in seconds before showing the Application Launch dialog to users.

Command Line

Command option: `--launch-showdialogafter secs`

Usage: Replace *secs* with the delay, measured in seconds.

In the following example, the Application Launch dialog is displayed after two seconds.

```
--launch-showdialogafter 2
```

“Launch Details” Pane

Usage: Select or deselect the check boxes.

Description

Attributes that control the initial display state of the Launch Details area of the Application Launch dialog, whether users can change it and whether to show the Launch Details area if an application startup fails.

If users cannot change the setting, the Showed by Default attribute determines whether the users see the application launch details.

Command Line

Command option: `--launch-details-initial hidden | shown`

Command option: `--launch-details-state enabled | disabled`

Command option: `--launch-details-showonerror 1 | 0`

Usage: Specify a valid option.

In the following example, the initial state of the Launch Details area is *hidden*. Users can change this setting. The Launch Details area is shown if the application fails to start.

```
--launch-details-initial hidden
--launch-details-state enabled
--launch-details-showonerror 1
```

Communication Tab

Settings on the Communication tab control connections between the client device, the SGD server, and application servers. They also control the resumability behavior for application sessions.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

This tab contains the following sections:

- Ports

This section contains the following attributes:

- [Unencrypted Connections Port](#)
- [Encrypted Connections Port](#)

- Application Sessions

This section contains the following attributes:

- [AIP Keepalive Frequency](#)
- [Timeout for User Session Resumability](#)
- [Timeout for General Resumability](#)

- Synchronization

This section contains the [Resource Synchronization Service](#) attribute.

- User Sessions

This section contains the [User Session Idle Timeout](#) attribute.

Unencrypted Connections Port

Usage: Type a port number in the field.

Description

The TCP port number used for *unencrypted* connections between client devices and SGD servers.

Open this port in your firewall to enable connections from users who have standard connections. Standard connections are connections that do not use SSL.

You must restart every SGD server in the array for changes to this attribute to take effect.

The default is TCP port 3144.

Command Line

Command option: `--array-port-unencrypted tcp-port`

Usage: Replace *tcp-port* with the port number to use for unencrypted connections.

In the following example, TCP port 3144 is used for unencrypted connections.

```
--array-port-unencrypted 3144
```

Encrypted Connections Port

Usage: Type a port number in the field.

Description

The TCP port number used for *encrypted* connections between client devices and SGD servers.

Open this port in your firewall to enable connections from users who have secure (SSL-based) connections to SGD.

You must restart every SGD server in the array for changes to this attribute to take effect.

The default is TCP port 5307.

Command Line

Command option: `--array-port-encrypted tcp-port`

Usage: Replace *tcp-port* with the port number to use for encrypted connections.

In the following example, TCP port 5307 is used for encrypted connections.

```
--array-port-encrypted 5307
```

AIP Keepalive Frequency

Usage: Type a time period, measured in seconds, in the field.

Description

Determines how often a keepalive message is sent to client devices during application sessions. The default value is 100 seconds.

Some Hypertext Transfer Protocol (HTTP) proxy servers close a connection if there is no activity on it. Using a keepalive ensures that a connection stays open.

Set this to 0 to disable keepalive messages.

This attribute is also used keep open connections between the SGD Client and the SGD server for client drive mapping.

You must restart every SGD server in the array for changes to this attribute to take effect.

Command Line

Command option: `--sessions-aipkeepalive secs`

Usage: Replace *secs* with the keepalive time period, measured in seconds.

In the following example, a keepalive message is sent to the client device every 100 seconds.

```
--sessions-aipkeepalive 100
```

Timeout for User Session Resumability

Usage: Type a timeout value, measured in minutes, in the field.

Description

For applications configured to be resumable during the user session, the length of time in minutes that a suspended application session is guaranteed to be resumable for if the connection to SGD is lost. Note that if the user logs out, the application sessions end. See the [Application Resumability](#) attribute.

After this period, the SGD server ends the session.

You can override this setting using the [Application Resumability: Timeout](#) attribute of an application.

Note – If an application is terminated because the SGD Client exits unexpectedly, the timeout is the timeout plus 20 minutes.

Changes to this attribute take effect immediately.

Command Line

Command option: `--sessions-timeout-session mins`

Usage: Replace *mins* with the timeout value, measured in minutes.

In the following example, the application session is resumable for 1440 minutes (24 hours).

```
--sessions-timeout-session 1440
```

Timeout for General Resumability

Usage: Type a timeout value, measured in minutes, in the field.

Description

For applications configured to be generally resumable, the length of time in minutes that a suspended application session is guaranteed to be resumable for after the user logs out or the connection to SGD is lost. See the [Application Resumability](#) attribute.

After this period the SGD server ends the session.

You can override this setting using the [Application Resumability: Timeout](#) attribute of an application.

Note – If an application is terminated because the SGD Client exits unexpectedly, the timeout is the timeout plus 20 minutes.

Changes to this attribute take effect immediately.

Command Line

Command option: `--sessions-timeout-always mins`

Usage: Replace *mins* with the timeout value, measured in minutes.

In the following example, the application session is resumable for 11500 minutes.

```
--sessions-timeout-always 11500
```

Resource Synchronization Service

Usage: Select or deselect the check box.

Description

Whether to enable replication of resources for the array.

If enabled, synchronization starts at a time determined by the [Daily Resource Synchronization Time](#) for each SGD server in the array.

Resource synchronization is enabled by default.

Changes to this attribute take effect immediately.

Command Line

Command option: `--array-resourcesync 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables resource synchronization for the array.

```
--array-resourcesync 0
```

User Session Idle Timeout

Usage: Type a timeout value, measured in seconds, in the field.

Description

User sessions are ended automatically if there has been no application session or webtop activity for the specified time period. The timeout applies to all SGD servers in the array.

The user session idle timeout is disabled by default. A setting of 0 turns off the feature.



Caution – Do not configure an idle timeout that is less than 300 seconds (five minutes).

Activity on the following devices has no effect on the idle timeout period:

- Serial ports
- Smart cards
- Audio

You must restart every SGD server in the array for changes to this attribute to take effect.

Command Line

Command option: `--webtop-session-idle-timeout secs`

Usage: Replace *secs* with the timeout value, measured in seconds.

In the following example, user sessions are ended automatically after being inactive for 1800 seconds, or 30 minutes.

```
--webtop-session-idle-timeout 1800
```

Performance Tab

Attributes on the Performance tab are used to specify the following load balancing settings:

- The method for selecting the SGD server used to host the application session
- The method for selecting the application server used to host the application

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Changes to these attributes take effect immediately.

Application Session Load Balancing

Usage: Choose an option.

Description

The algorithm used at application start time to choose the SGD server in the array that hosts the application session. In other words, the method used to choose where to run the Protocol Engine when a user starts an application.

Select the Server Hosting the User Session option to choose the SGD server in the array that is hosting the user session.

Command Line

Command option: `--sessions-loadbalancing-algorithm algorithm`

Usage: Replace *algorithm* with the load balancing algorithm to use for application sessions.

The following algorithms are available:

- **Server Hosting the User Session –**
`.../_beans/com.sco.tta.server.loadbalancing.tier2.LocalLoadBalancingPolicy`
- **Least CPU Usage –**
`.../_beans/com.sco.tta.server.loadbalancing.tier2.CpuLoadBalancingPolicy`
- **Fewest Application Sessions –**
`.../_beans/com.sco.tta.server.loadbalancing.tier2.SessionLoadBalancingPolicy`

The following example specifies that the SGD server hosting the user session is used to host the application session.

```
--sessions-loadbalancing-algorithm \  
.../_beans/com.sco.tta.server.loadbalancing.tier2.LocalLoadBalancingPolicy
```

Application Load Balancing

Usage: Select an option.

Description

The default algorithm SGD uses to choose the best application server to run the application. The server is selected from those defined on the application object's Hosting Application Servers tab.

This attribute is only used if the value of the application object's [Application Load Balancing](#) attribute is not set to Override Global Setting.

Select one of the following settings:

- **Most Free Memory.** Choose the application server with the most free memory.
- **Least CPU Usage.** Choose the application server with the most central processing unit (CPU) idle time.
- **Fewest Applications.** Choose the application server that is running the fewest application sessions through SGD. This is the default setting.

Note – To use the Most Free Memory and Least CPU Usage algorithms, you must install the SGD Enhancement Module on the application server.

Command Line

Command option: `--launch-loadbalancing-algorithm cpu | memory | sessions`

Usage: Specify a valid option.

In the following example, the application server with the fewest application sessions is used to run the application.

```
--launch-loadbalancing-algorithm sessions
```

Client Device Tab

Attributes on the Client Device tab are settings for the user's client device. This tab controls the use of client device features for applications displayed through SGD.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

This tab contains the following sections:

- Client Drive Mapping

This section contains the following attributes:

- [Windows Client Drive Mapping](#)
- [Unix Client Drive Mapping](#)
- [Dynamic Drive Mapping](#)

- Audio

This section contains the following attributes:

- [Windows Audio](#)
- [Unix Audio](#)

- Other Features

This section contains the following attributes:

- [Smart Card](#)
- [Serial Port Mapping](#)
- [Copy and Paste](#)
- [Client's Clipboard Security Level](#)
- [Time Zone Map File](#)

- Profile Editing

This section contains the [Editing](#) attribute.

Windows Client Drive Mapping

Usage: Select or deselect the check box.

Description

Whether to enable client drive mapping (CDM) for applications running on Windows application servers.

Changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--array-windowscdm 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables Windows CDM for the array.

```
--array-windowscdm 1
```

Unix Client Drive Mapping

Usage: Select or deselect the check box.

Description

Whether to enable CDM for applications running on UNIX or Linux platform application servers.

To use UNIX platform CDM, the Oracle Secure Global Desktop Enhancement Module (SGD Enhancement Module) must be installed and running on the application server.

If you enable UNIX platform CDM, CDM services only become available when you restart all SGD servers in the array. To manually start CDM services without restarting the array, run the `tarantella start cdm` command on all SGD servers in the array.

If you disable UNIX platform CDM, the CDM processes only stop when you restart all SGD servers in the array. To manually stop CDM services without restarting the array, run the `tarantella stop cdm` command on all SGD servers in the array.

Changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--array-unixcdm 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables UNIX platform CDM for the array.

```
--array-unixcdm 1
```

Dynamic Drive Mapping

Usage: Select or deselect the check box.

Description

Whether to enable dynamic drive mapping for the array. This feature enables “hot plugging” of removable storage devices, such as Universal Serial Bus (USB) drives.

By default, dynamic drive mapping is enabled for an SGD array.

To use dynamic drive mapping, CDM must be enabled for the array. This means that the [Windows Client Drive Mapping](#) or [Unix Client Drive Mapping](#) attribute must be enabled.

To use dynamic drive mapping for UNIX and Linux platform application servers, the Oracle Secure Global Desktop Enhancement Module (SGD Enhancement Module) must be installed and running on the application server.

Changes to this attribute only take effect when you restart all the SGD servers in the array.

Command Line

Command option: `--array-dyndevice 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables dynamic drive mapping for the array.

```
--array-dyndevice 0
```

Windows Audio

Usage: Select or deselect the check box.

Description

Whether to enable Windows audio services for the array.

To play audio for Windows applications, audio redirection must be enabled on the Windows Terminal Server.

Changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--array-audio 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables Windows audio services for the array.

```
--array-audio 0
```

Windows Audio Sound Quality

Usage: Select an option.

Description

The sample rate of the audio data.

Adjusting the audio quality increases or decreases the amount of audio data sent.

By default, SGD uses Medium Quality Audio.

The sample rates are as follows:

- **Low Quality Audio** – 8 kHz.
- **Medium Quality Audio** – 22.05 kHz.
- **High Quality Audio** – Same as Medium Quality Audio. This is a Terminal Services restriction.

Command Line

Command option: `--array-audio-quality low | medium | high`

Usage: Specify an audio quality setting.

The following example specifies medium quality audio for Windows audio services.

```
--array-audio-quality medium
```

Unix Audio

Usage: Select or deselect the check box.

Description

Whether to enable UNIX platform audio services for the array.

UNIX platform audio is only available for X applications. The audio module of the SGD Enhancement Module must be installed and running on the application server.

Changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--array-unixaudio 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables UNIX platform audio services for the array.

```
--array-unixaudio 0
```

Unix Audio Sound Quality

Usage: Select an option.

Description

The sample rate of the audio data.

Adjusting the audio quality increases or decreases the amount of audio data sent.

By default, SGD uses Medium Quality Audio.

The sample rates are as follows:

- **Low Quality Audio** – 8 kHz

- **Medium Quality Audio** – 22.05 kHz
- **High Quality Audio** – 44.1 kHz

Command Line

Command option: `--array-unixaudio-quality low | medium | high`

Usage: Specify an audio quality setting.

The following example specifies medium quality audio for UNIX platform audio services.

```
--array-unixaudio-quality medium
```

Smart Card

Usage: Select or deselect the check box.

Description

Whether to enable smart card services for the array.

To use smart cards, smart card device redirection must be enabled on the Windows Terminal Server.

Changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--array-scard 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables smart card services for the array.

```
--array-scard 1
```

Serial Port Mapping

Usage: Select or deselect the check box.

Description

Whether to enable access to serial ports for the array.

By default, access to serial ports is enabled.

Access to serial ports for individual users can be enabled and disabled using the [Serial Port Mapping](#) attribute for organization, organizational unit or user profile objects.

Changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--array-serialport 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables access to serial ports for the array.

```
--array-serialport 1
```

Copy and Paste

Usage: Select or deselect the check box.

Description

Whether to allow copy and paste operations for Windows and X application sessions for the array.

By default, copy and paste is allowed.

Copy and paste operations for individual users can be enabled and disabled using the [Copy and Paste](#) attribute for organization, organizational unit, or user profile objects.

Changes to this attribute only take effect for new application sessions.

Command Line

Command option: `--array-clipboard-enabled 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables copy and paste for Windows and X application sessions.

```
--array-clipboard-enabled 1
```

Client's Clipboard Security Level

Usage: Type a number in the field.

Description

The security level for the SGD Client.

Used to control copy and paste operations between Windows or X application sessions and applications running on the client device.

The security level can be any positive integer. The higher the number, the higher the security level. The default security level is 3.

Changes to this attribute only take effect for new application sessions.

Command Line

Command option: `--array-clipboard-clientlevel num`

Usage: Replace *num* with a positive integer that specifies the security level.

The following example specifies a client clipboard security level of 3.

```
--array-clipboard-clientlevel 3
```

Time Zone Map File

Usage: Type the file name in the field.

Description

A file that contains mappings between UNIX platform client device and Windows application server time zone names.

Command Line

Command option: `--xpe-tzmapfile filename`

Usage: Replace *filename* with the path to the time zone map file.

In the following example, a time zone map file is specified.

```
--xpe-tzmapfile "%INSTALLDIR%/etc/data/timezonemap.txt"
```

Editing

Usage: Select or deselect the check box.

Description

Whether to allow users to edit their own profiles for use with the SGD Client.

By default, profile editing is enabled.

If profile editing is disabled, it is disabled for *all* users, including SGD Administrators. However, SGD Administrators can still create and edit profiles using the Profile Editor application.

Profile editing for individual users can be enabled and disabled using the [Client Profile Editing](#) attribute for organization, organizational unit, or user profile objects.

Changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--array-editprofile 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables user profile editing for the array.

```
--array-editprofile 1
```

Printing Tab

Attributes on the Printing tab control printing from Windows applications.

The settings on this tab are default settings, which can be overridden by the following attributes:

- Client Printing: Override (`--userprintingconfig`) attribute for an organization, organizational unit, or user profile object.
- Client Printing: Override (`--appprintingconfig`) attribute for a Windows application object. If you configure a Windows application object, this overrides the printing configuration for organization, organizational unit, or user profile objects.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Client Printing

Usage: Select an option.

Description

Controls the client printers users can print to from Windows application.

By default, users can print to all their client printers.

If you select the No Printer option, you can still use an SGD Portable Document Format (PDF) printer.

Changes to this attribute take effect for new user sessions.

If SGD is configured so you can only print to the client's default printer and you want to print to a different printer, log out of SGD. Then change the default printer and log in to SGD again.

Command Line

Command option: `--printing-mapprinters 2 | 1 | 0`

Usage: Specify one of the following options:

- 2 – Allow users to print to all client printers
- 1 – Allow users to print to the client's default printer
- 0 – No client printers available

The following example enables the user to print to all client printers from a Windows application.

```
--printing-mapprinters 2
```

Universal PDF Printer

Usage: Select or deselect the check box.

Description

Enables users to print from a Windows application using the SGD Universal PDF printer.

When a user prints to the Universal PDF printer, the print job is converted into a PDF file and is printed on the user's client device.

This is enabled by default.

Changes to this attribute take effect for new user sessions.

Command Line

Command option: `--printing-pdfenabled 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables printing from Windows applications to the SGD Universal PDF printer.

```
--printing-pdfenabled 1
```

Make Universal PDF Printer the Default

Usage: Select or deselect the check box.

Description

Sets the SGD Universal PDF printer as the client's default printer when printing from a Windows application.

When a user prints to the Universal PDF printer, the print job is converted into a PDF file and is printed on the user's client device.

This attribute is only available if the Universal PDF printer is enabled.

By default, the Universal PDF printer is not the default printer.

Changes to this attribute take effect for new user sessions.

Command Line

Command option: `--printing-pdfisdefault 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, the SGD Universal PDF printer is set to be the client's default printer.

```
--printing-pdfisdefault 1
```

Universal PDF Viewer

Usage: Select or deselect the check box.

Description

Enables users to print from a Windows application using the SGD Universal PDF Viewer printer.

When a user prints to the Universal PDF Viewer printer, the print job is converted into a PDF file and can be viewed, saved, or printed on the user's client device.

This attribute is enabled by default.

Changes to this attribute take effect for new user sessions.

Command Line

Command option: `--printing-pdfviewerenabled 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables printing from Windows applications to the SGD Universal PDF Viewer printer.

```
--printing-pdfviewerenabled 1
```

Make Universal PDF Viewer the Default

Usage: Select or deselect the check box.

Description

Sets the SGD Universal PDF Viewer printer as the client's default printer when printing from a Windows application.

When a user prints to the Universal PDF Viewer printer, the print job is converted into a PDF file and can be viewed, saved or printed on the user's client device.

This attribute is only available if Universal PDF Viewer is enabled.

By default, the Universal PDF Viewer printer is not the default printer.

Changes to this attribute take effect for new user sessions.

Command Line

Command option: `--printing-pdfviewerisdefault 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, the SGD Universal PDF Viewer printer is set to be the client's default printer.

```
--printing-pdfviewerisdefault 0
```

Postscript Printer Driver

Usage: Type the printer driver name in the field.

Description

The name of the printer driver to use for SGD PDF printing. This printer driver must be installed on every Windows application server used with SGD.

The printer driver must be a PostScript™ printer driver. The default is HP Color LaserJet 2800 Series PS.

The name of the printer driver must match the name of the printer driver installed on the Windows application server exactly. Pay particular attention to the use of capitals and spaces. The `/opt/tarantella/etc/data/default.printerinfo.txt` file contains all the common printer driver names, ordered by manufacturer. To avoid errors, copy and paste the driver name from this file.

Changes to this attribute take effect for new user sessions.

Command Line

Command option: `--printing-pdfdriver driver_name`

Usage: Replace *driver_name* with the PDF printer driver name.

In the following example, an HP Laserjet 4000 driver is used for PDF printing.

```
--printing-pdfdriver "HP Laserjet 4000 Series PS"
```

Security Tab

Attributes on the Security tab are global security attributes which apply to all SGD servers in the array.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

New Password Encryption Key

Usage: Select or deselect the check box.

Description

Whether to generate a new encryption key for the password cache when an SGD server is restarted.

If a new encryption key is generated, the existing password cache is preserved and encrypted with the new key.

Command Line

Command option: `--security-newkeyonrestart 1 | 0`

Usage: Specify 1 (true) or 0 (false).

In the following example, a new encryption key for the password cache is not generated when an SGD server is restarted.

```
--security-newkeyonrestart 0
```

Timeout for Print Name Mapping

Usage: Type a timeout value, measured in seconds, in the field.

Description

The period of time an entry in the print name mapping table is retained. This table is used to ensure that users can print from an application and then exit the application, without losing the print job.

The timer starts counting when the user closes the last application on the application server.

Set the timeout value to be greater than the maximum delay between choosing to print from an application and the printer responding.

If you change this value, all existing expiry timeouts are reset. Changes take effect immediately.

To flush the table, type in 0 and click Apply. You can then set the timeout to the required value.

To display the table, use the `tarantella print status --namemapping` command.

Command Line

Command option: `--security-printmappings-timeout seconds`

Usage: Replace *seconds* with the timeout value, measured in seconds.

In the following example, the print name mapping table is retained for 1800 seconds (30 minutes).

```
--security-printmappings-timeout 1800
```

Connection Definitions

Usage: Select or deselect the check box.

Description

Whether to take note of the [Connections](#) attribute when a user logs in to SGD.

Select the check box, or set the command line option to 1, if you are using the Connections attribute for user profile, organizational unit, or organization objects.

Deselect the check box if SGD security services are not enabled.

If SGD security services are enabled, connections are secure unless the check box is selected *and* some connections are defined otherwise.

Deselecting the check box enables users to log in more quickly.

Changes to this attribute take effect immediately.

Command Line

Command option: `--security-applyconnections 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables checking of connections for SGD log ins.

```
--security-applyconnections 0
```


X Authorization for X Display

Usage: Select or deselect the check box.

Description

Whether to secure all SGD X displays using X authorization. This prevents users from accessing X displays they are not authorized to access.

X authorization is enabled by default.

To use X authorization, `xauth` must be installed on the application server.

If X authorization is enabled, SGD checks the standard locations for the `xauth` binary. Extra configuration might be needed if the binary is in a nonstandard location.

Changes to this attribute take effect immediately.

Note – This attribute only secures the X display between the SGD server and the application server.

Command Line

Command option: `--security-xsecurity 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables X authorization.

```
--security-xsecurity 1
```

Monitoring Tab

Settings on the Monitoring tab are used to configure system message log filters and enable billing services.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Log Filter

Usage: Type log filter definitions in the field. Press the Return key to add new entries.

Description

This attribute specifies which diagnostic messages are logged and a destination file or handler for log messages.

The attribute contains multiple values, each of the form:

`component/subcomponent/severity:destination`

Use the wildcard (*) to match multiple components, subcomponents and severities.

Valid destinations are a file name or the name of a plug-in log handler.

File names can include the placeholder %%PID%%, which is substituted with a process ID.

Changes to this attribute take effect immediately.

Command Line

Command option: `--array-logfilter filter...`

Usage: Replace *filter...* with a list of log filter definitions. Separate each *filter* definition with a space. Quote any filters that contain wildcards (*), to stop your shell from expanding them.

The following example specifies a log filter that stores all warnings and error messages for the SGD server to a .log file.

`--array-logfilter */*/*error:jserver%%PID%%_error.log`

Billing Service

Usage: Select or deselect the check box.

Description

Whether to enable billing services for the array.

This might use significant additional disk space on SGD servers in the array.

If enabled, you can use `tarantella query billing` to analyze the billing logs.

You must restart an SGD server for billing services to start.

Command Line

Command option: `--array-billingservices 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables billing services for the array.

```
--array-billingservices 0
```

Resilience Tab

Attributes on the Resilience tab are used to configure settings for array resilience. Array resilience is used when the primary SGD server in an array becomes unavailable.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Array Failover

Usage: Select or deselect the check box.

Description

Whether to enable array failover for the array. By default, array failover is disabled.

Changes to this attribute take effect immediately.

Command Line

Command option: `--array-failoverenabled 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables array failover for the SGD array.

```
--array-failoverenabled 1
```

Monitor Interval

Usage: Type a time period, measured in seconds, in the field.

Description

The length of time, in seconds, between operations used to monitor the array. The default value is 60 seconds.

This attribute is used with the [Monitor Attempts](#) attribute to determine the time period before array failover is started.

Changes to this attribute take effect immediately.

Command Line

Command option: `--array-monitortime secs`

Usage: Replace *secs* with the array monitor interval, measured in seconds.

The following example sets the array monitor interval to 30 seconds.

```
--array-monitortime 30
```

Monitor Attempts

Usage: Type a number in the field.

Description

The number of consecutive array monitoring operations that must fail before array failover is started. The default value is 10.

This attribute is used with the [Monitor Interval](#) attribute to determine the time period before array failover is started.

Changes to this attribute take effect immediately.

Command Line

Command option: `--array-maxmonitors num`

Usage: Replace *num* with the maximum number of array monitor attempts.

The following example sets the maximum number of array monitor attempts to 5.

```
--array-maxmonitors 5
```

Find Primary Interval

Usage: Type a time period, measured in seconds, in the field.

Description

When array failover begins, a new primary server is selected from the backup primaries list. This attribute configures the length of time, in seconds, between attempts to contact the new primary server. The default value is 60 seconds.

This attribute is used with the [Find Primary Attempts](#) attribute to determine the timeout period for contacting a new primary server. If the contact operation fails after this time period, the next server in the backup primaries list is used.

Changes to this attribute take effect immediately.

Command Line

Command option: `--array-resubmitfindprimarywait secs`

Usage: Replace *secs* with the find primary interval, measured in seconds.

The following example sets the find primary interval to 30 seconds.

```
--array-resubmitfindprimarywait 30
```

Find Primary Attempts

Usage: Type a number in the field.

Description

When array failover begins, a new primary server is selected from the backup primaries list. This attribute configures the maximum number of attempts made to contact the new primary server. The default value is 3.

This attribute is used with the [Find Primary Interval](#) attribute to determine the timeout period for contacting a new primary server. If the contact operation fails after this time period, the next server in the backup primaries list is used.

Changes to this attribute take effect immediately.

Command Line

Command option: `--array-resubmitfindprimarymax num`

Usage: Replace *num* with the maximum number of find primary attempts.

The following example sets the maximum number of find primary attempts to 5.

```
--array-resubmitfindprimarymax 5
```

Action When Failover Ends

Usage: Select an option.

Description

Determines what happens when the original primary server becomes available after array failover.

The options are as follows:

- **Restore original primary** – The original primary server, and any attached secondary servers, rejoin the array. The original array formation is restored automatically. This is the default setting.
- **Do not restore original array** – The original primary server, and any attached secondary servers, do not rejoin the array. The original primary server, and any attached secondary servers, remain in the array formed during the failover stage.
- **Restore array with a new primary** – The original primary server, and any attached secondary servers, rejoin the array as secondary servers. To use this option, all secondaries must be in the same array after failover. Otherwise, the recovery operation fails and the array is not restored.

Command Line

Command option: `--array-primaryreturnaction accept | ignore | acceptsecondary`

Usage: Specify the primary return action setting.

The following example specifies that the original primary server, and any attached secondary servers, do not rejoin the array after array failover.

```
--array-primaryreturnaction ignore
```

Backup Primaries

Usage: Use the Backup Primaries table to manage the backup primaries list. Use the New and Delete buttons to add and remove servers in the Backup Primaries table. Order the servers using the Move Up and Move Down buttons. Use the Reload button to refresh the table.

Description

The backup primaries list is a list of secondary servers that can be promoted to primary server during array failover. When you build an array, the backup primaries list is created automatically. If you add a secondary server to the array, an entry is added at the end of the list. If you remove a secondary server from the array, the entry for the server is removed from the list.

Entries in the backup primaries list are in order of priority, with the highest priority secondary server at the top of the list.

Clicking the New button displays the Available Secondaries table, a list of secondary servers in the array that are not on the backup primaries list. To add a secondary server in the Available Secondaries table to the backup primaries list, select the server and click Add.

Command Line

On the command line, use the `tarantella array` commands to manage the backup primaries list. See [“The tarantella array Command” on page 675](#).

Caches Tab

The Caches tab is where you can view, edit, and manage the caches used by SGD for authentication.

The Caches tab includes the following tabs:

- [“Passwords Tab” on page 488](#)
- [“Tokens Tab” on page 490](#)

Passwords Tab

Usage: Use the Password Cache table to manage entries in the password cache.

Description

The Passwords tab lists *all* password cache entries for the SGD array.

Use the New button to add a password cache entry, using the Create New Password Cache Entry page.

Use the Edit button to edit an entry in the password cache, or the Delete button to remove an entry from the password cache.

Use the Reload button to refresh the Password Cache table.

Use the Search field to search for entries in the Password Cache table. You can use the "*" wildcard in your search string. Typing a search string of *name* is equivalent to searching for "**name**" and returns any match of the search string. The number of results returned by a search is limited to 150, by default.

Adding Entries to the Password Cache

When you create a new password cache entry, it is important that you enter a valid name in the User Identity or Server fields on the Create New Password Cache Entry page. The Administration Console supports several ways that you can enter a name in the User Identity or Server field, as follows:

- **Browse button.** If the selected User Identity Type option is Local or LDAP/Active Directory, you can use the Browse button next to the User Identity or Server field to browse for object names. Using the Browse button in this way avoids errors when typing in object names.
- **Full Name.** Type the *full name* into the field. For example, you can type in the fully qualified name for an application server from the local repository as follows:

```
.../_ens/o=appservers/cn=boston
```

- **Partial Name.** Type a *partial name*, without the namespace prefix, in the field. Depending on the selected User Identity Type option, the Administration Console adds the relevant namespace prefix when the password cache entry is saved.

For example, if you select UNIX (User/Groups) as the User Identity Type and type `o=organization/cn=Indigo Jones` in the field, the Administration Console creates the password cache entry using the name `.../_user/o=organization/cn=Indigo Jones`.

The Administration Console adds the `.../_user` namespace prefix when the password cache entry is saved.

The following table shows the namespace prefixes that the Administration Console adds for the selected User Identity Type option.

| User Identity Type | Namespace Prefix |
|---------------------------|--|
| Local | <code>.../_ens</code> |
| UNIX (User/Groups) | <code>.../_user</code> |
| Windows Domain Controller | <code>.../_wns</code> |
| LDAP/Active Directory | <code>.../service/sco/tta/ldapcache</code> |

| User Identity Type | Namespace Prefix |
|--------------------|--------------------------------|
| SecurID | .../service/sco/tta/securid |
| Anonymous | None |
| Third Party | .../service/sco/tta/thirdparty |

If you specify a partial name in the Server field, the Administration Console adds the .../_ens/o=appservers namespace prefix when the password cache entry is saved.

LDAP names must be typed in using the SGD naming format. The following example shows a partial name for a user identity from an LDAP repository:

```
dc=com/dc=example/cn=indigo-jones
```

This name is converted to the correct LDAP format when the password cache entry is saved, as follows:

```
.../_service/sco/tta/ldapcache/cn=indigo-jones,dc=example,dc=com
```

Command Line

On the command line, use the `tarantella passcache` commands to list, add, and delete password cache entries. See [“The tarantella passcache Command” on page 762](#).

Tokens Tab

Usage: Use the Token Cache table to manage entries in the token cache.

Description

The Tokens tab is used to manage tokens used for the authentication token authentication mechanism. This authentication mechanism is used when the SGD Client is in Integrated mode.

The Tokens tab lists *all* token cache entries for the SGD array.

Use the Delete button to delete a token from the token cache.

Use the Reload button to refresh the Token Cache table.

Use the Search field to search for entries in the Token Cache table. You can use the `"*"` wildcard in your search string. Typing a search string of `name` is equivalent to searching for `"*name*"` and returns any match of the search string. The number of results returned by a search is limited to 150, by default.

Command Line

On the command line, use the `tarantella tokencache` commands to list and delete token cache entries. See [“The tarantella tokencache Command” on page 836](#).

Secure Global Desktop Server Settings

Secure Global Desktop servers are machines running Oracle Secure Global Desktop (SGD) software. By adding at least one other server you create an array. An array can distribute load between its servers and therefore increase reliability. One server in the array is the *primary server*, which is responsible for replicating configuration data. Other servers in the array are called *secondary servers*.

Use the Secure Global Desktop Server Settings tab to set up an SGD server array, or to configure settings for a particular SGD server.

This chapter includes the following topics:

- [“Secure Global Desktop Servers Tab” on page 494](#)
- [“General Tab” on page 495](#)
- [“Security Tab” on page 498](#)
- [“Performance Tab” on page 500](#)
- [“Protocol Engines Tab” on page 504](#)
- [“Character Protocol Engine Tab” on page 505](#)
- [“X Protocol Engine Tab” on page 507](#)
- [“Execution Protocol Engine Tab” on page 512](#)
- [“Channel Protocol Engine Tab” on page 515](#)
- [“Print Protocol Engine Tab” on page 517](#)
- [“Audio Protocol Engine Tab” on page 518](#)
- [“IO Protocol Engine Tab” on page 519](#)
- [“User Sessions Tab” on page 520](#)
- [“Application Sessions Tab” on page 521](#)

Secure Global Desktop Servers Tab

The Secure Global Desktop Servers tab gives you an overview of the current status of each SGD server in the array, including how many user and application sessions each server is hosting.

SGD server information is shown in the Secure Global Desktop Server List table.

If you click the name of a server in the Secure Global Desktop Server List table a series of tabs are displayed. The tabs are used to view and change the configuration for the server.

The following tabs are shown:

- [General Tab](#)
- [Security Tab](#)
- [Performance Tab](#)
- [Protocol Engines Tab](#)
- [User Sessions Tab](#)
- [Application Sessions Tab](#)

The Secure Global Desktop Server List Table

The number of SGD servers in the array is indicated in brackets at the top of the table.

The Add Server button adds an SGD server to the array. The SGD server is added as a secondary server.

If you select a secondary server in the table, the Make Primary button makes the selected server the primary server in the SGD array.

The Remove Server button removes the selected SGD server from the array. The selected SGD server must be a secondary server.

You update the Secure Global Desktop Server List table by clicking the Reload button.

The Secure Global Desktop Server List table includes the following information for each SGD server in the array:

- **Server.** Domain Name System (DNS) name of the SGD server.
- **Type.** Whether the server is a primary or secondary server.

- **Status.** Server status, for example, whether the server is running.
- **Start Time.** When the server was last started.
- **Accepting Connections.** Whether the server is accepting standard connections, secure connections or both types of connection. Secure connections use the Secure Sockets Layer (SSL) to encrypt data. Standard connections do not encrypt data.
- **User Sessions.** The current number of user sessions on this server. The numbers of user sessions using standard and secure connections are shown.
- **Application Sessions.** The current number of application sessions on this server, including those that are currently suspended. The numbers of graphical application sessions and terminal-based application sessions are shown.

Command Line

From the command line, use the `tarantella array` commands to add servers to the SGD array, remove servers from the SGD array, make a secondary server the primary server, or view information about the SGD array. See [“The `tarantella array` Command” on page 675](#).

General Tab

Attributes on the General tab are general attributes for a particular SGD server.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Changes to these attributes take effect immediately.

External DNS Names

Usage: Type the external DNS names of this server in the field. Press the Return key after each name definition.

Description

The external DNS names of this server.

This attribute enables you to use different names, depending on the Internet Protocol (IP) address of the client.

Only change this setting if this server is known by different names on the network, for example, inside and outside a firewall.

Each name has the following format:

IP-pattern:DNS name

IP-pattern is a regular expression, or a subnet mask, matching a client IP address. For example, 192.168.10.*, or 192.168.10.0/24.

If this server only has one name, use one line matching all clients. For example, *:www.indigo-insurance.com.

The order of the names is important. The DNS name for the *first* matching IP pattern is used.

Note – You must restart the SGD server for a change to this setting to take effect.

Command Line

Command option: --server-dns-external *IP-pattern:dns-name*

Usage: Replace *IP-pattern* with a regular expression for the client IP addresses. Replace *dns-name* with the external DNS name of the server. Use a comma to separate multiple DNS names.

In the following example, a DNS name of boston.indigo-insurance.com is used for clients with an IP address in the 192.168.10.* range. All other clients use a DNS name of www.indigo-insurance.com.

```
--server-dns-external "192.168.10.*:boston.indigo-insurance.com, \
*:www.indigo-insurance.com"
```

User Login

Usage: Select or deselect the check box.

Description

Whether to allow users to log in to this SGD server.

To “decommission” an SGD server, deselect the check box. No users can log in and no new application sessions can start. Users currently logged in to this server, or with application sessions hosted on this server, are not affected. Users can log in to another SGD server in the array and resume application sessions hosted on this server.

Users are redirected to the web page defined by the [Redirection URL](#) attribute. Typically, you set this to another SGD server in the array.

Command Line

Command option: `--server-login enabled | disabled`

Usage: Specify enabled or disabled.

In the following example, user logins are disabled for the SGD host.

```
--server-login disabled
```

Redirection URL

Usage: Type a redirection Uniform Resource Locator (URL) in the field.

Description

If the SGD server does not allow users to log in, client devices are redirected to this URL.

If the attribute is not set, client devices are redirected to a page telling users that they cannot log in.

Command Line

Command option: `--server-redirectionurl url`

Usage: Replace *url* with the address of a web page to redirect to.

The following example specifies a redirection URL of `www.indigo-insurance.com`.

```
--server-redirectionurl "www.indigo-insurance.com"
```

Security Tab

Attributes on the Security tab are security attributes for a particular SGD server in the array.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Changes to these settings take effect immediately.

Connection Types

Usage: Select the check box for each connection type you want to make available to users.

Description

The possible connection types available to users.

Secure connections use SSL to encrypt transmissions.

For standard connections, transmissions are not encrypted.

Command Line

Command option: `--security-connectiontypes types`

Usage: Specify the connection types to use.

Valid settings are `std` (standard connections only), `ssl` (secure connections only), or `std,ssl` (both standard and secure connections).

The following example specifies standard connections only.

```
--security-connectiontypes std
```

SSL Accelerator Support

Usage: Select or deselect the check box.

Description

Select the check box to enable support for an external SSL accelerator.

Selecting this check box enables the SGD SSL daemon to accept plain text traffic and pass it on to the SGD server as if it was SSL traffic it had decoded.

Command Line

Command option: `--security-acceptplaintext 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables SSL accelerator support.

```
--security-acceptplaintext 1
```

Firewall Forwarding URL

Usage: Type a URL in the field.

Description

The absolute URL to forward all web server traffic not related to SGD.

Use this feature if you plan to run SGD on the same port as your web server, so that you do not have to open any additional ports in your firewall.

Command Line

Command option: `--security-firewallurl server-url`

Usage: Replace *server-url* with a firewall forwarding URL.

The following example specifies a URL to forward all non-SGD web traffic to.

```
--security-firewallurl https://127.0.0.1:443
```

Performance Tab

Use attributes on the Performance tab to tune the SGD server.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Maximum Simultaneous Requests

Usage: Type a number in the field.

Description

The maximum number of requests the server processes simultaneously.

As a rough guide, set this to the number of central processing units (CPUs) multiplied by 4.

Too high a setting might degrade performance.

Changes to this attribute take effect immediately.

Command Line

Command option: `--tuning-maxrequests num`

Usage: Replace *num* with the maximum number of simultaneous requests.

The following example sets the maximum number of simultaneous requests to 7.

```
--tuning-maxrequests 7
```

Maximum Simultaneous User Sessions

Usage: Type a number in the field.

Description

The maximum number of simultaneous user sessions. A user session is defined as a connection between an SGD Client and the SGD server.

Once the limit is reached, connections are refused.

Too high a setting might degrade performance.

Changes to this attribute take effect immediately.

Command Line

Command option: `--tuning-maxconnections num`

Usage: Replace *num* with the maximum number of simultaneous user sessions.

The following example sets the maximum number of simultaneous user sessions to 1000.

```
--tuning-maxconnections 1000
```

Maximum File Descriptors

Usage: Type a number in the field.

Description

The maximum number of open file descriptors allowed.

Increasing this value increases the number of simultaneous connections that can be handled.

This value affects all SGD server components.

Too high a setting might degrade performance.

Changes to this attribute take effect when the server restarts.

Command Line

Command option: `--tuning-maxfiledescriptors num`

Usage: Replace *num* with the maximum number of open file descriptors.

The following example sets the maximum number of open file descriptors to 4096.

```
--tuning-maxfiledescriptors 4096
```

JVM Size

Usage: Type numbers in the fields.

Description

These attributes control the size and expansion rate of the memory allocated to the SGD server's Java™ Runtime Environment (JRE™). The following attributes are available:

- The amount of memory, in megabytes, to allocate initially for the SGD server's Java Virtual Machine (JVM™). Set this to no greater than the amount of random access memory (RAM) on the host.
- A scaling factor, expressed as a percentage, used to increase the amount of JVM software memory dynamically when needed.
- An absolute maximum size in megabytes, that is never exceeded.

Too high a setting might degrade performance.

Changes to this attribute take effect when the server or JVM software restarts.

Command Line

Command option: `--tuning-jvm-initial MB`

Usage: Replace *MB* with the initial memory allocation for the JVM software, in megabytes.

Command option: `--tuning-jvm-scale percent`

Usage: Replace *percentage* with a dynamic scaling factor, expressed as a percentage.

Command option: `--tuning-jvm-max MB`

Usage: Replace *MB* with the maximum memory allocation for the JVM software, in megabytes.

The following examples set the initial JVM software size to 58 megabytes. The amount of JVM software memory can be scaled up to 150% when needed. The maximum JVM software size is set to 512 megabytes.

```
--tuning-jvm-initial 58
--tuning-jvm-scale 150
--tuning-jvm-max 512
```

Daily Resource Synchronization Time

Usage: Type a number in the field.

Description

When to start resource synchronization each day, if enabled for the array.

Use the server's local time zone.

Express the time in 24-hour clock format. For example, use 16:00 for 4 p.m.

Changes to this attribute take effect immediately.

Command Line

Command option: `--tuning-resourcesync-time hh:mm`

Usage: Replace *hh:mm* with the time, in 24-hour clock format.

The following example sets the resource synchronization time to 4:00 (4 a.m.)

```
--tuning-resourcesync-time 4:00
```

Load Balancing Groups

Usage: Type the load balancing groups for this SGD server in the field.

Description

This attribute is a string identifying the load balancing group for an SGD server in an array. This information can be used for application load balancing.

This attribute is used to enable optimal bandwidth usage. SGD servers are chosen from the same load balancing groups as application servers, where possible.

Leave this attribute blank unless your array spans a wide area network (WAN), or includes slow links, and you are using load balancing.

More than one string is allowed, but this slows application launch.

If used, set this attribute on all SGD servers in the array, and all application server objects in the organizational hierarchy.

Command Line

Command option: `--server-location location`

Usage: Replace *location* with a string identifying the load balancing group for the SGD server in the array.

The following example specifies a location of boston.

```
--server-location boston
```

Protocol Engines Tab

The Protocol Engines tab contains several tabs where you can change settings for the Protocol Engines running on the SGD server.

A Protocol Engine is an SGD software component that runs on an SGD server. Protocol Engines emulate native protocols, such as X11 and Microsoft Remote Desktop Protocol (RDP), and communicate with application servers. Protocol Engines also send display data to the client device using Adaptive Internet Protocol (AIP).

You can change settings for the following Protocol Engines:

- Character
- X
- Execution
- Channel
- Print
- Audio
- Smart Card

Character Protocol Engine Tab

Use the attributes on the Character Protocol Engine tab to tune terminal emulator processes.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Changes to these attributes take effect for new Protocol Engines only. Existing Protocol Engines are not affected.

Maximum Sessions

Usage: Type a number in the field.

Description

The maximum number of application sessions each Character Protocol Engine handles.

More Character Protocol Engines are started to meet demand.

Command Line

Command option: `--cpe-maxsessions num`

Usage: Replace *num* with the maximum number of application sessions.

The following example specifies a maximum application sessions setting of 20 for each Character Protocol Engine.

```
--cpe-maxsessions 20
```

Exit Timeout

Usage: Type a number in the field.

Description

The length of time, in seconds, a Character Protocol Engine process continues to run without any active connections.

Command Line

Command option: `--cpe-exitafter secs`

Usage: Replace *num* with the time period, measured in seconds.

In the following example, the Protocol Engine exits after 60 seconds if there are no active connections.

```
--cpe-exitafter 60
```

Command-Line Arguments

Usage: Type command-line arguments in the field.

Description

Any arguments to the Protocol Engine. For example, the name of a log file.

Only change this setting if Technical Support ask you to.

Command Line

Command option: `--cpe-args args`

Usage: Replace *args* with the arguments to pass to the Protocol Engine.

The following example specifies an error log file for the Protocol Engine.

```
--cpe-args cpeerror.log
```

X Protocol Engine Tab

Use attributes on the X Protocol Engine tab to tune graphical emulator processes.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Changes to these attributes take effect for new Protocol Engines only. Existing Protocol Engines are not affected.

Monitor Resolution

Usage: Type a number in the field.

Description

The default monitor resolution, in dots per inch, to assume.

You can override this value using an application's [Monitor Resolution](#) attribute.

Command Line

Command option: `--xpe-monitorresolution dpi`

Usage: Replace *dpi* with the monitor resolution, in dots per inch.

The following example specifies a monitor resolution of 96 dots per inch.

```
--xpe-monitorresolution 96
```

Font Path

Usage: Type path names for the fonts directories in the field.

Description

Directories on the SGD host containing the fonts used by the X Protocol Engine.

Font paths are listed in search order.

Use `%%INSTALLDIR%%` to represent the SGD installation directory.

You can include font servers, for example, `tcp/boston:7000`.

Command Line

Command option: `--xpe-fontpath fontpath`

Usage: Replace *fontpath* with a list of font directories. Separate each directory in the font path with a comma `,`.

The following example specifies a list of font directories used by the X Protocol Engine.

```
--xpe-fontpath %%INSTALLDIR%%/etc/fonts/misc,\n%%INSTALLDIR%%/etc/fonts/TTF, %%INSTALLDIR%%/etc/fonts/Type1
```

RGB Database

Usage: Type the path name of the RGB database file in the field.

Description

Full path name on the SGD host of the RGB database used by the X Protocol Engine to resolve color names to RGB values.

Use `%%INSTALLDIR%%` to represent the SGD installation directory.

Command Line

Command option: `--xpe-rgbdatabase file`

Usage: Replace *file* with the full path name of the RGB database file.

The following example specifies the RGB database used by the X Protocol Engine.

```
--xpe-rgbdatabase %%INSTALLDIR%%/etc/data/rgb.txt
```

Keyboard Map

Usage: Select the required keyboard map option. For custom keyboard maps, type a file name in the field.

Description

The default keyboard map to use for graphical applications.

To specify a keyboard map based on a locale, do one of the following:

- Select LANG Variable to use the locale of the SGD server
- Select Client's Input Locale to use the locale of the client device

The actual keyboard map used is determined using the `/opt/tarantella/etc/data/keymaps/xlocales.txt` file.

Note – You can use the `*` and `?` wildcards in the `xlocales.txt` file to support a wide range of input locales. See the `xlocales.txt` file for details.

Alternatively, you can type a filename to always use a particular keyboard map.

You can override this for each user with the user profile object's [Keyboard Map](#) attribute.

Command Line

Command option: `--xpe-keymap lang | client-locale | file`

Usage: Specify a valid setting. For custom keyboard maps, replace *file* with the full path name of the keyboard map file.

In the following example, a keyboard map based on the locale of the client device is used.

```
--xpe-keymap client-locale
```

Client Window Size

Usage: Type numbers for horizontal and vertical display sizes, in pixels, in the fields.

Description

The maximum expected horizontal and vertical display resolution for client devices connecting to this server.

Use these attributes to tune the Client Window Management value of the [Window Type](#) attribute.

These attributes only apply for applications with Window Type set to Client Window Management. Use them to avoid clipping problems.

Command Line

Command option: `--xpe-cwm-maxwidth pixels`

Command option: `--xpe-cwm-maxheight pixels`

Usage: Replace *pixels* with a value for maximum display width or maximum display height.

The following example specifies a maximum display size of 1280 x 960 pixels.

```
--xpe-cwm-maxwidth 1280  
--xpe-cwm-maxheight 960
```

Session Start Timeout

Usage: Type a number in the field.

Description

How long the X Protocol Engine waits for X applications to connect, in seconds.

Command Line

Command option: `--xpe-sessionstarttimeout seconds`

Usage: Replace *seconds* with a timeout value, in seconds.

The following example specifies a timeout value of 60 seconds when starting an X session.

```
--xpe-sessionstarttimeout 60
```

Maximum Sessions

Usage: Type a number in the field.

Description

The maximum number of application sessions each X Protocol Engine handles.
More X Protocol Engines are started to meet demand.

Command Line

Command option: `--xpe-maxsessions num`

Usage: Replace *num* with the maximum number of application sessions.

The following example specifies a maximum sessions setting of 20 for each X Protocol Engine.

```
--xpe-maxsessions 20
```

Exit Timeout

Usage: Type a number in the field.

Description

The length of time, in seconds, an X Protocol Engine process continues to run without any active connections.

Command Line

Command option: `--xpe-exitafter secs`

Usage: Replace *num* with the time period, measured in seconds.

In the following example, the Protocol Engine exits after 60 seconds if there are no active connections.

```
--xpe-exitafter 60
```

Command-Line Arguments

Usage: Type command-line arguments in the field.

Description

Any arguments to the Protocol Engine. For example, the name of a log file.

Only change this setting if Technical Support ask you to.

Command Line

Command option: `--xpe-args args`

Usage: Replace *args* with the arguments to pass to the Protocol Engine.

The following example specifies an error log file for the Protocol Engine.

```
--xpe-args xpeerror.log
```

Execution Protocol Engine Tab

Use the attributes on the Execution Protocol Engine tab to tune application startup processes.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Changes to these attributes take effect for new Protocol Engines only. Existing Protocol Engines are not affected.

Maximum Sessions

Usage: Type a number in the field.

Description

The maximum number of application sessions each Execution Protocol Engine handles.

More Execution Protocol Engines are started to meet demand.

Command Line

Command option: `--execpe-maxsessions num`

Usage: Replace *num* with the maximum number of application sessions.

The following example specifies a maximum sessions setting of 10 for each Execution Protocol Engine.

```
--execpe-maxsessions 10
```

Exit Timeout

Usage: Type a number in the field.

Description

The length of time, in seconds, an Execution Protocol Engine process continues to run without any active connections.

Command Line

Command option: `--execpe-exitafter secs`

Usage: Replace *secs* with the time period, measured in seconds.

In the following example, the Protocol Engine exits after 60 seconds if there are no active connections.

```
--execpe-exitafter 60
```

Login Script Directory

Usage: Type a directory path name in the field.

Description

The directory on the SGD host where login scripts are stored.

Use `%%INSTALLDIR%%` to represent the SGD installation directory.

If an application object's [Login Script](#) attribute uses a relative path name, for example `unix.exp`, this directory is assumed.

Only change this setting if Technical Support ask you to.

Command Line

Command option: `--execpe-scriptdir dir`

Usage: Replace *dir* with the path name for the login script directory.

In the following example, the login script directory for a default SGD installation is `/opt/tarantella/var/serverresources/expect`.

```
--execpe-scriptdir %%INSTALLDIR%%/var/serverresources/expect
```

Command-Line Arguments

Usage: Type command-line arguments in the field.

Description

Any arguments to the Protocol Engine. For example, the name of a log file.

Only change this setting if Technical Support ask you to.

Command Line

Command option: `--execpe-args args`

Usage: Replace *args* with the arguments to pass to the Protocol Engine.

The following example specifies an error log file for the Protocol Engine.

```
--execpe-args execpeerror.log
```

Channel Protocol Engine Tab

Use the attributes on the Channel Protocol Engine tab to tune SGD channel processes. The SGD channel is used to detect information about the client. For example, to detect client drives or audio devices.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Changes to these attributes take effect for new Protocol Engines only. Existing Protocol Engines are not affected.

Packet Compression

Usage: Choose a compression setting option.

Description

Whether a Channel Protocol Engine uses data compression on a client connection.

Select On Slow Connection to enable the Channel Protocol Engine to compress data if the connection is slow.

Command Line

Command option: `--chpe-compression auto | always | never`

Usage: Specify a valid compression setting.

The following example enables data compression for slow client connections only.

```
--chpe-compression auto
```

Packet Compression Threshold

Usage: Type a compression threshold value, measured in bytes, in the field.

Description

The smallest size of network packet that a Channel Protocol Engine can compress.

Command Line

Command option: `--chpe-compressionthreshold bytes`

Usage: Replace *bytes* with a compression threshold setting, in bytes.

In the following example, a minimum packet size of 256 bytes is specified. Network packets smaller than this value are not compressed.

```
--chpe-compressionthreshold 256
```

Exit Timeout

Usage: Type a number in the field.

Description

The length of time, in seconds, a Channel Protocol Engine process continues to run without any active connections.

Command Line

Command option: `--chpe-exitafter secs`

Usage: Replace *secs* with the time period, measured in seconds.

In the following example, the Protocol Engine exits after 60 seconds if there are no active connections.

```
--chpe-exitafter 60
```

Print Protocol Engine Tab

Use the attributes on the Print Protocol Engine tab to tune SGD printing processes.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Changes to these attributes take effect for new Protocol Engines only. Existing Protocol Engines are not affected.

Packet Compression

Usage: Choose a compression setting option.

Description

Whether a Print Protocol Engine uses data compression on a client connection.

Select On Slow Connection to enable the Print Protocol Engine to compress data if the connection is slow.

Command Line

Command option: `--ppe-compression auto | always | never`

Usage: Specify a valid compression setting.

The following example enables data compression for slow client connections.

```
--ppe-compression auto
```

Packet Compression Threshold

Usage: Type a compression threshold value, measured in bytes, in the field.

Description

The smallest size of file that a Print Protocol Engine can compress.

Command Line

Command option: `--ppe-compressionthreshold bytes`

Usage: Replace *bytes* with a compression threshold setting, in bytes.

In the following example, a minimum file size of 4096 bytes is specified. Print files smaller than this value are not compressed.

```
--ppe-compression 4096
```

Exit Timeout

Usage: Type a number in the field.

Description

The length of time, in seconds, a Print Protocol Engine process continues to run without any active connections.

Command Line

Command option: `--ppe-exitafter secs`

Usage: Replace *secs* with the time period, measured in seconds.

In the following example, the Protocol Engine exits after 240 seconds if there are no active connections.

```
--ppe-exitafter 240
```

Audio Protocol Engine Tab

Use the attributes on the Audio Protocol Engine tab to tune SGD audio processes.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Changes to these attributes take effect for new Protocol Engines only. Existing Protocol Engines are not affected.

Packet Compression

Usage: Choose a compression setting option.

Description

Whether an Audio Protocol Engine uses data compression on a client connection.

By default, compression is off. This is to avoid unnecessarily compressing audio data that might already be compressed.

Select On Slow Connection to enable the Audio Protocol Engine to compress data if the connection is slow.

Command Line

Command option: `--audiope-compression auto | always | never`

Usage: Specify a valid compression setting.

The following example enables data compression for slow client connections only.

```
--audiope-compression auto
```

IO Protocol Engine Tab

Use the attributes on the IO Protocol Engine tab to tune SGD smart card, serial port, and Windows client drive mapping (CDM) processes.

From the command line, use the `tarantella config list` command to list these settings, and the `tarantella config edit` command to edit these settings.

Changes to these attributes take effect for new Protocol Engines only. Existing Protocol Engines are not affected.

Packet Compression

Usage: Choose a compression setting option.

Description

Whether an IO Protocol Engine uses data compression on a client connection.

Select On Slow Connection to enable the IO Protocol Engine to compress data if the connection is slow.

Command Line

Command option: `--iope-compression auto | always | never`

Usage: Specify a valid compression setting.

The following example enables data compression for slow client connections.

```
--iope-compression auto
```

User Sessions Tab

The User Sessions tab enables you to view and manage user sessions for the SGD server. A user session represents a user that is connected to an SGD server.

User session information is shown in the User Session List table.

The User Session List Table

The User Session List table shows details of user sessions for the SGD server.

The number of user sessions is indicated in brackets at the top of the table.

The User Session List table includes the following information for each user session:

- **User Identity.** A unique identifier for the user.
- **User Profile.** A profile that defines configuration settings and the applications available to the user.
- **Secure Global Desktop Server.** The name of the SGD server hosting the user session.
- **Login Time.** When the user logged in to the SGD server.

Use the Search options to search the User Session List table. The number of results returned by a search is limited to 150, by default. When searching for a User Identity or User Profile, you can use the "*" wildcard in your search string. Typing a search string of *name* is equivalent to searching for "**name**" and returns any match of the search string. The Start Time search option enables you to search for user sessions that were started between a range of times that you specify. The format of each time specified is *yyyy/mm/dd hh:mm:ss*. Use one of the following formats to specify the range:

- starttime - endtime

For example, 2010/05/11 08:00:00 - 2010/05/11 17:00:00.

Searches for user sessions that were started between the specified times.

- starttime-

For example, 2010/05/11 08:00:00-.

Searches for user sessions that were started on or after the specified time.

- -endtime

For example, -2010/05/11 08:00:33.

Searches for user sessions that were started before the specified time.

To show more details about a user session, select the check box for the user session in the User Session List table and click the View Details button.

To end a user session, select the check box for the user session in the User Session List table and click the End button.

To end all user sessions, click the Select Items Currently Displayed icon to select all user sessions and click the End button.

You can update the User Session List table by clicking the Reload button.

Command Line

From the command line, use the `tarantella webtopsession` command to list user session details, and end user sessions. See ["The tarantella webtopsession Command" on page 850](#).

Application Sessions Tab

The Application Sessions tab enables you to view and manage application sessions for the SGD server.

Application session information is shown in the Application Session List table.

The Application Session List Table

The Application Session List table shows details of application sessions for the SGD server.

The number of application sessions is indicated in brackets at the top of the table.

The Application Session List table includes the following information for each application session:

- **User Identity.** A unique identifier for the user.
- **User Profile.** A profile that defines configuration settings and the applications available to the user.
- **Secure Global Desktop Server.** The name of the SGD server hosting the application session.
- **Application Server.** The name of the application server hosting the application.
- **Application.** The name of the application.
- **Start Time.** When the application was started.
- **Status.** Current state of the application, for example, whether the application is running or suspended.

You can use the Search options to search the Application Session List table. The number of results returned by a search is limited to 150, by default. When searching for a User Identity, User Profile, or Application Server, you can use the "*" wildcard in your search string. Typing a search string of *name* is equivalent to searching for "**name**" and returns any match of the search string. The Start Time search option enables you to search for application sessions that were started between a range of times that you specify. The format of each time specified is *yyyy/mm/dd hh:mm:ss*. Use one of the following formats to specify the range:

- **starttime - endtime**

For example, 2010/05/11 08:00:00 - 2010/05/11 17:00:00.

Searches for application sessions that were started between the specified times.

- **starttime-**

For example, 2010/05/11 08:00:00-.

Searches for application sessions that were started on or after the specified time.

- **-endtime**

For example, -2010/05/11 08:00:33.

Searches for application sessions that were started before the specified time.

To show more details about an application session, select the check box for the application session in the Application Session List table and click the View Details button.

To end an application session, select the check box for the application session in the Application Session List table and click the End button.

To end all application sessions, click the Select Items Currently Displayed icon to select all application sessions and click the End button.

You can update the Application Session List table by clicking the Reload button.

Shadowing an application session enables you and the user to interact with the application simultaneously. To shadow an application session, select the check box for the application session in the Application Session List table and click the Shadow button.

Note – In some countries, it is illegal to shadow a user without their knowledge. It is your responsibility to comply with the law.

Shadowing is not supported for character applications or suspended applications. A warning message is shown if you attempt to shadow either of these applications.

Command Line

From the command line, use the `tarantella emulatorsession` command to list application session details, shadow application sessions, and end application sessions. See [“The `tarantella emulatorsession` Command” on page 692](#).

User Profiles, Applications, and Application Servers

Oracle Secure Global Desktop (SGD) represents users, resources, and the structure of your organization as *objects* in a directory. Different types of object have different configuration settings, known as *attributes*.

The object types used by SGD and their attributes are described in this chapter. This chapter includes the following topics:

- [“SGD Objects” on page 525](#)
- [“Attributes Reference” on page 545](#)

SGD Objects

The supported object types in SGD are as follows:

- 3270 Application Object
- 5250 Application Object
- Application Server Object
- Character Application Object
- Directory: Organization Object
- Directory: Organizational Unit Object
- Directory (Light): Active Directory Container Object
- Directory (Light): Domain Component Object
- Document Object
- Dynamic Application Object
- Dynamic Application Server Object

- Group Object
- User Profile Object
- Windows Application Object
- X Application Object

3270 Application Object

Use a 3270 application object to give a 3270 application to users.

SGD uses the third-party TeemTalk for Unix emulator for 3270 applications. See the TeemTalk for Unix User's Guide supplied with SGD for details.

To create a 3270 application object, use the Administration Console or the `tarantella object new_3270app` command.

In the Administration Console, the configuration settings for 3270 application objects are divided into a series of tabs.

The *General tab* contains the attributes that control the name and the icon used when creating links for users. The attributes on the General tab are as follows:

- Name
- Comment
- Icon
- Hints

The *Launch tab* contains the attributes that control how the application is started and whether application sessions can be suspended and resumed. The attributes on the Launch tab are as follows:

- Arguments for Command
- Connection Method
- Connection Method: SSH Arguments
- Login Script
- Environment Variables
- Number of Sessions
- Application Resumability
- Application Resumability: Timeout
- Keep Launch Connection Open
- Session Termination
- Window Close Action

The *Presentation tab* contains the attributes that control how the application displays to users. The attributes on the Presentation tab are as follows:

- Window Type
- Window Manager
- Window Size: Client's Maximum Size
- Window Size: Scale to Fit Window
- Window Size: Width
- Window Size: Height
- Window Color
- Window Color: Custom Color

The *Performance tab* contains the attributes for optimizing the performance of the application. The attributes on the Performance tab are as follows:

- Command Compression
- Command Execution
- Delayed Updates
- Graphics Acceleration
- Interlaced Images
- Share Resources Between Similar Sessions

The *Client Device tab* contains the attributes that control how the user's client device interacts with the application. The attributes on the Client Device tab are as follows:

- Keyboard Map: Locked
- Euro Character
- Copy and Paste
- Copy and Paste: Application's Clipboard Security Level
- Middle Mouse Timeout
- Monitor Resolution

The *Third-Party Emulator tab* contains the attributes for the third-party TeemTalk for Unix emulator. The attributes on the Third-Party Emulator tab are as follows:

- Server Address
- Server Port
- Connection Closed Action
- Window Size: Maximized
- Menu Bar
- 'File' and 'Settings' Menus
- Displayed Soft Buttons

- [Foreground Color](#)
- [Background Color](#)
- [Keyboard Type](#)

The *Assigned User Profiles tab* lists the user profile objects that can run the application. See [Assigned User Profiles Tab](#).

The *Application Sessions tab* lists the running and suspended application sessions for the application. See [Application Sessions Tab](#).

5250 Application Object

Use a 5250 application object to give a 5250 application to users.

SGD uses the third-party TeemTalk for Unix emulator for 5250 applications. See the TeemTalk for Unix User's Guide supplied with SGD for details.

To create a 5250 application object, use the Administration Console or the `tarantella object new_5250app` command.

In the Administration Console, the configuration settings for 5250 application objects are divided into a series of tabs.

The *General tab* contains the attributes that control the name and the icon used when creating links for users. The attributes on the General tab are as follows:

- [Name](#)
- [Comment](#)
- [Icon](#)
- [Hints](#)

The *Launch tab* contains the attributes that control how the application is started and whether application sessions can be suspended and resumed. The attributes on the Launch tab are as follows:

- [Arguments for Command](#)
- [Connection Method](#)
- [Connection Method: SSH Arguments](#)
- [Login Script](#)
- [Environment Variables](#)
- [Number of Sessions](#)
- [Application Resumability](#)
- [Application Resumability: Timeout](#)
- [Keep Launch Connection Open](#)

- Session Termination
- Window Close Action

The *Presentation tab* contains the attributes that control how the application displays to users. The attributes on the Presentation tab are as follows:

- Window Type
- Window Manager
- Window Size: Client's Maximum Size
- Window Size: Scale to Fit Window
- Window Size: Width
- Window Size: Height
- Window Color
- Window Color: Custom Color

The *Performance tab* contains the attributes for optimizing the performance of the application. The attributes on the Performance tab are as follows:

- Command Compression
- Command Execution
- Delayed Updates
- Graphics Acceleration
- Interlaced Images
- Share Resources Between Similar Sessions

The *Client Device tab* contains the attributes that control how the user's client device interacts with the application. The attributes on the Client Device tab are as follows:

- Keyboard Map: Locked
- Euro Character
- Copy and Paste
- Copy and Paste: Application's Clipboard Security Level
- Middle Mouse Timeout
- Monitor Resolution

The *Third-Party Emulator tab* contains the attributes for the third-party TeemTalk for Unix emulator. The attributes on the Third-Party Emulator tab are as follows:

- Server Address
- Server Port
- Connection Closed Action
- Window Size: Maximized
- Menu Bar

- 'File' and 'Settings' Menus
- Displayed Soft Buttons
- Foreground Color
- Background Color
- Keyboard Type

The *Assigned User Profiles tab* lists the user profile objects that can run the application. See [Assigned User Profiles Tab](#).

The *Application Sessions tab* lists the running and suspended application sessions for the application. See [Application Sessions Tab](#).

Application Server Object

Use an application server object to represent an application server that is used to run applications through SGD.

Application server objects are used with application load balancing. If you assign two or more application server objects to an application object, SGD chooses the application server to use, based on the load across the application servers.

To create an application server object, use the Administration Console or the `tarantella object new_host` command.

In the Administration Console, the configuration settings for application server objects are divided into a series of tabs.

The *General tab* contains the attributes that control the designation and application authentication for the application server. The attributes on the General tab are as follows:

- Name
- Comment
- Address
- Application Start
- User Assignment
- Maximum Count
- Domain Name
- Password Cache Usage
- Prompt Locale

The *Performance tab* contains the attributes for optimizing the performance of applications. See [Load Balancing Groups](#).

The *Hosted Applications tab* lists the applications hosted on the application server. See [Hosted Applications Tab](#).

The *Application Sessions tab* lists the running and suspended application sessions for the application server. See [Application Sessions Tab](#).

The *Passwords tab* lists the entries of the password cache for the application server. See [Passwords Tab](#).

Character Application Object

Use a character application object to give a VT420, Wyse 60, or SCO Console character application to users.

Character application objects support VT420, Wyse 60, or SCO Console character applications. The [Emulation Type](#) attribute determines the type of application.

To create a character application object, use the Administration Console or the `tarantella object new_charapp` command.

In the Administration Console, the configuration settings for character application objects are divided into a series of tabs.

The *General tab* contains the attributes that control the name and the icon used when creating links for users. The attributes on the General tab are as follows:

- [Name](#)
- [Comment](#)
- [Icon](#)
- [Hints](#)

The *Launch tab* contains the attributes that control how the application is started and whether application sessions can be suspended and resumed. The attributes on the Launch tab are as follows:

- [Application Command](#)
- [Arguments for Command](#)
- [Connection Method](#)
- [Connection Method: SSH Arguments](#)
- [Login Script](#)
- [Environment Variables](#)
- [Answerback Message](#)
- [Number of Sessions](#)
- [Application Resumability](#)

- Application Resumability: Timeout
- Window Close Action

The *Presentation tab* contains the attributes that control how the application displays to users. The attributes on the Presentation tab are as follows:

- Window Type
- Emulation Type
- Terminal Type
- Window Size: Client's Maximum Size
- Window Size: Width
- Window Size: Height
- Window Size: Columns
- Window Size: Lines
- Font Family
- Font Size: Fixed Font Size
- Font Size
- Border Style
- Cursor
- Attribute Map
- Color Map
- Scroll Style
- Status Line
- Line Wrapping

The *Performance tab* contains the attributes for optimizing the performance of the application. The attributes on the Performance tab are as follows:

- Application Load Balancing
- Command Compression

The *Client Device tab* contains the attributes that control how the user's client device interacts with the application. The attributes on the Client Device tab are as follows:

- Keyboard Map
- Keyboard Codes Modification
- Numpad Codes Modification
- Cursor Key Codes Modification
- Escape Sequences
- Code Page

The *Hosting Application Servers tab* lists the application servers that are configured to host the application. See [Hosting Application Servers Tab](#).

The *Assigned User Profiles tab* lists the user profile objects that can run the application. See [Assigned User Profiles Tab](#).

The *Application Sessions tab* lists the running or suspended application sessions for the application. See [Application Sessions Tab](#).

Directory: Organization Object

Use an organization object for things that apply to your organization as a whole.

Organization objects are always at the top of the organizational hierarchy.

Organization objects can contain organizational unit (OU) or user profile objects.

To create an organization object, use the Administration Console or the `tarantella object new_org` command.

In the Administration Console, the configuration settings for organization objects are divided into a series of tabs.

The *General tab* contains the attributes that control the name of the organization. The attributes on the General tab are as follows:

- [Name](#)
- [Comment](#)

The *Client Device tab* contains the attributes that control how the user's client device interacts with the application. The attributes on the Client Device tab are as follows:

- [Client Profile Editing](#)
- [Copy and Paste](#)
- [Serial Port Mapping](#)
- [Client Drive Mapping](#)

The *Printing tab* contains the attributes for users printing from Windows applications. The attributes on the Printing tab are as follows:

- [Client Printing: Override](#)
- [Client Printing](#)
- [Universal PDF Printer](#)
- [Make Universal PDF Printer the Default](#)
- [Universal PDF Viewer](#)
- [Make Universal PDF Viewer the Default](#)

- [Postscript Printer Driver](#)

The *Security tab* contains attributes that define the connections that are allowed between the client device and the SGD server. See [Connections](#).

The *Assigned Applications tab* lists the applications that are available to users in the organization. See [Assigned Applications Tab](#).

Directory: Organizational Unit Object

Use an organizational unit (OU) object to distinguish different departments, sites, or teams in your organization.

An OU can be contained in an organization or a domain component object.

To create an OU object, use the Administration Console or the [tarantella object new_orgunit](#) command.

In the Administration Console, the configuration settings for OU objects are divided into a series of tabs.

The *General tab* contains the attributes that control the name of the OU. The attributes on the General tab are as follows:

- [Name](#)
- [Comment](#)

The *Client Device tab* contains the attributes that control how the user's client device interacts with the application. The attributes on the Client Device tab are as follows:

- [Client Profile Editing](#)
- [Copy and Paste](#)
- [Serial Port Mapping](#)
- [Client Drive Mapping](#)

The *Printing tab* contains the attributes for users printing from Windows applications. The attributes on the Printing tab are as follows:

- [Client Printing: Override](#)
- [Client Printing](#)
- [Universal PDF Printer](#)
- [Make Universal PDF Printer the Default](#)
- [Universal PDF Viewer](#)
- [Make Universal PDF Viewer the Default](#)
- [Postscript Printer Driver](#)

The *Security tab* contains attributes that define the connections that are allowed between the client device and the SGD server. See [Connections](#).

The *Assigned Applications tab* lists the applications that are available to users in the organizational unit. See [Assigned Applications Tab](#).

Directory (Light): Active Directory Container Object

Use an Active Directory container object to replicate your Microsoft Active Directory structure within the SGD organizational hierarchy.

Active Directory container objects are similar to organizational unit objects, but do not include additional SGD-specific attributes or allow you to assign applications. This is why they are called Directory (light) objects.

An Active Directory container object can be contained in an Organization, an Organizational Unit, or a domain component object.

To create an Active Directory container object, use the Administration Console or the `tarantella object new_container` command.

In the Administration Console, the configuration settings for an Active Directory container object are divided into a series of tabs.

The General tab contains the attributes that control the name of the Active Directory container. See [Name](#).

Directory (Light): Domain Component Object

Use a domain component object to replicate a directory structure, usually a Microsoft Active Directory structure, within the SGD organizational hierarchy.

Domain component objects are similar to organization objects, but do not include additional SGD-specific attributes or allow you to assign applications. That is why they are called Directory (light) objects.

Domain component objects can only appear at the top of the organizational hierarchy, or within another domain component object.

Domain component objects can contain organizational unit, domain component, Active Directory container, or user profile objects

To create a domain component object use the Administration Console or the `tarantella object new_dc` command.

In the Administration Console, the configuration settings for domain component objects are divided into a series of tabs.

The *General tab* contains the attributes that control the name of the domain component. See [Name](#).

Document Object

Use a document object to give a document to users.

A document object can refer to any Uniform Resource Locator (URL). This can be any document on the web, including Oracle Open Office software documents, or Adobe® Acrobat software files. A document can also refer to a web application.

It is the user's *client device* that actually fetches the URL and so firewall or other security measures might prevent a user from accessing a URL.

To create a document object, use the Administration Console or the `tarantella object new_doc` command.

In the Administration Console, the configuration settings for document objects are divided into a series of tabs.

The *General tab* contains the attributes that control the name and the icon used when creating links for users. The attributes on the General tab are as follows:

- [Name](#)
- [Comment](#)
- [Icon](#)
- [Hints](#)

The *Launch tab* contains the URL that is displayed when users click the link for the document. See [URL](#).

The *Presentation tab* contains the attribute that controls how the document displays to users. See [Window Type: New Browser Window](#).

The *Assigned User Profiles tab* lists the user objects that can access the document. See [Assigned User Profiles Tab](#).

Group Object

Use a group object to associate groups of applications with a user profile, organizational unit, or organization, or to associate similar application servers for application load balancing.

Group objects are not the same as organizational units. Applications and application servers can only belong to one organizational unit, but can be a member of many different groups.

Members of a group can be moved or renamed without affecting group membership.

Group objects can be added to the following tabs for an object.

- **Assigned Applications tab.** Use this tab to assign a group of applications to a user profile, organizational unit, or organization object. The group members are shown recursively, but not the group itself. See [Assigned Applications Tab](#).
- **Hosting Application Servers tab.** Use this tab to assign a group of application servers to an application object. The group members are used recursively for application server load balancing. See [Hosting Application Servers Tab](#).

To create a group object, use the Administration Console or the `tarantella object new_group` command.

In the Administration Console, the configuration settings for group objects are divided into a series of tabs.

The *General tab* contains the attributes that control the name of the group. The attributes on the General tab are as follows:

- [Name](#)
- [Comment](#)

The *Members tab* is used to display and edit the members of the group object. See [Members Tab](#).

The *Assigned User Profiles tab* lists the user profile objects that can run the applications in the group. See [Assigned User Profiles Tab](#).

The *Hosted Applications tab* lists the applications hosted on the application servers in the group. See [Hosted Applications Tab](#).

User Profile Object

Use a user profile object to represent a user in your organization, and give that user access to applications.

Depending on the authentication mechanisms used, users might be able to log in to SGD even if they do not have a user profile object.

To use inheritance, create user profile objects within organizational units. This makes administration easier and more efficient, see [Inherit Assigned Applications from Parent](#).

To create a user profile object, use the Administration Console or the [tarantella object new_person](#) command.

In the Administration Console, the configuration settings for user profile objects are divided into a series of tabs.

The *General tab* contains user naming attributes for user designation and authentication. The attributes on the General tab are as follows:

- [Name](#)
- [Comment](#)
- [Surname](#)
- [Login](#)
- [Login: Multiple](#)
- [Login Name](#)
- [Email Address](#)
- [Domain Name](#)

The *Performance tab* contains the attributes that control the user's bandwidth limit. See [Bandwidth Limit](#).

The *Client Device tab* contains the attributes that control how the user's client device interacts with the application. The attributes on the Client Device tab are as follows:

- [Client Profile Editing](#)
- [Copy and Paste](#)
- [Keyboard Map](#)
- [Serial Port Mapping](#)
- [Client Drive Mapping](#)

The *Printing tab* contains the attributes for users printing from Windows applications. The attributes on the Printing tab are as follows:

- [Client Printing: Override](#)
- [Client Printing](#)
- [Universal PDF Printer](#)
- [Make Universal PDF Printer the Default](#)
- [Universal PDF Viewer](#)
- [Make Universal PDF Viewer the Default](#)
- [Postscript Printer Driver](#)

The *Security tab* contains attributes that define the connections that are allowed between the client device and the SGD server. See [Connections](#).

The *Assigned Applications tab* lists the applications that are available to the user. See [Assigned Applications Tab](#).

The *Passwords tab* lists the entries in the password cache for the user. See [Passwords Tab](#).

The *Tokens tab* lists the authentication tokens for the user. See [Tokens Tab](#).

The *User Sessions tab* lists the active user sessions for the user. See [User Sessions Tab](#).

The *Application Sessions tab* lists the running and suspended application sessions for the user. See [Application Sessions Tab](#).

Dynamic Application Object

Use a dynamic application object with dynamic launch to enable users to select an application to run.

To create a dynamic application object, use the Administration Console or the `tarantella object new_dynamicapp` command.

In the Administration Console, the configuration settings for dynamic application objects are divided into a series of tabs.

The *General tab* contains the attributes that control the name of the dynamic application. The attributes on the General tab are as follows:

- [Name](#)
- [Comment](#)
- [Icon](#)

The *Mappings tab* lists the mappings for a dynamic application. This tab contains mappings between the type field and an application. See [Mappings Tab](#).

The *Assigned User Profiles tab* lists the user profile objects that can run the application. See [Assigned User Profiles Tab](#).

Dynamic Application Server Object

Use a dynamic application server object with dynamic launch to enable users to select the application server that runs an application.

To create a dynamic application server object, use the Administration Console or the `tarantella object new_host` command.

In the Administration Console, the configuration settings for dynamic application server objects are divided into a series of tabs.

The *General tab* contains the attributes that control the designation and application authentication for the dynamic application server. Attributes for broker configuration are also included. The attributes on the General tab are as follows:

- [Name](#)
- [Comment](#)
- [Password Cache Usage](#)
- [Virtual Server Broker Class](#)
- [Virtual Server Broker Parameters](#)

The *Hosted Applications tab* lists the applications assigned to the dynamic application server. See [Hosted Applications Tab](#).

The *Application Sessions tab* lists the running and suspended application sessions for the dynamic application server. See [Application Sessions Tab](#).

The *Passwords tab* lists the entries of the password cache for the dynamic application server. See [Passwords Tab](#).

Windows Application Object

Use a Windows application object to give a Microsoft Windows graphical application to users.

To create a Windows application object, use the Administration Console or the `tarantella object new_windowsapp` command.

In the Administration Console, the configuration settings for Windows application objects are divided into a series of tabs.

The *General tab* contains the attributes that control the name and the icon used when creating links for users. The attributes on the General tab are as follows:

- [Name](#)
- [Comment](#)
- [Icon](#)
- [Hints](#)

The *Launch tab* contains the attributes that control how the application is started and whether application sessions can be suspended and resumed. The attributes on the Launch tab are as follows:

- [Application Command](#)

- Arguments for Command
- Working Directory
- Local Client Launch
- SGD Remote Desktop Client
- Console Mode
- Arguments
- Domain Name
- Login Script
- Number of Sessions
- Application Resumability
- Application Resumability: Timeout
- Window Close Action

The *Presentation tab* contains the attributes that control how the application displays to users. The attributes on the Presentation tab are as follows:

- Window Type
- SWM Local Window Hierarchy
- Kiosk Mode Escape
- Window Manager
- Window Size: Client's Maximum Size
- Window Size: Scale to Fit Window
- Window Size: Width
- Window Size: Height
- Color Depth
- Desktop Wallpaper
- Full Window Drag
- Menu Animations
- Theming
- Cursor Shadow
- Cursor Settings
- Font Smoothing

The *Performance tab* contains the attributes for optimizing the performance of the application. The attributes on the Performance tab are as follows:

- Application Load Balancing
- Command Compression
- Command Execution

- [Delayed Updates](#)
- [Graphics Acceleration](#)
- [Interlaced Images](#)

The *Client Device tab* contains the attributes that control how the user's client device interacts with the application. The attributes on the Client Device tab are as follows:

- [Keyboard Map: Locked](#)
- [Window Management Keys](#)
- [Euro Character](#)
- [Copy and Paste: Application's Clipboard Security Level](#)
- [Middle Mouse Timeout](#)
- [Monitor Resolution](#)
- [Remote Audio](#)
- [Client Drive Mapping](#)

The *Printing tab* contains the attributes for users printing from Windows applications. The attributes on the Printing tab are as follows:

- [Printer Preference Caching](#)
- [Client Printing: Override](#)
- [Client Printing](#)
- [Universal PDF Printer](#)
- [Make Universal PDF Printer the Default](#)
- [Universal PDF Viewer](#)
- [Make Universal PDF Viewer the Default](#)
- [Postscript Printer Driver](#)

The *Hosting Application Servers tab* lists the application servers hosting the application. See [Hosting Application Servers Tab](#).

The *Assigned User Profiles tab* lists the user profile objects that can run the application. See [Assigned User Profiles Tab](#).

The *Application Sessions tab* lists the running and suspended application sessions for the application. See [Application Sessions Tab](#).

X Application Object

Use an X application object to give an X11 graphical application to users.

To create an X application object, use the Administration Console or the `tarantella object new_xapp` command.

In the Administration Console, the configuration settings for X application objects are divided into a series of tabs.

The *General tab* contains the attributes that control the name and the icon used when creating links for users. The attributes on the General tab are as follows:

- Name
- Comment
- Icon
- Hints

The *Launch tab* contains the attributes that control how the application is started and whether application sessions can be suspended and resumed. The attributes on the Launch tab are as follows:

- Application Command
- Arguments for Command
- Connection Method
- Connection Method: SSH Arguments
- Allow SSH Downgrade
- X Security Extension
- Login Script
- Environment Variables
- Number of Sessions
- Application Resumability
- Application Resumability: Timeout
- Keep Launch Connection Open
- Session Termination
- Window Close Action

The *Presentation tab* contains the attributes that control how the application displays to users. The attributes on the Presentation tab are as follows:

- Window Type
- Kiosk Mode Escape
- Window Manager
- Window Size: Client's Maximum Size
- Window Size: Scale to Fit Window
- Window Size: Width

- Window Size: Height
- Window Color
- Window Color: Custom Color
- Color Depth

The *Performance tab* contains the attributes for optimizing the performance of the application. The attributes on the Performance tab are as follows:

- Application Load Balancing
- Command Compression
- Command Execution
- Delayed Updates
- Graphics Acceleration
- Interlaced Images
- Color Quality
- Share Resources Between Similar Sessions

The *Client Device tab* contains the attributes that control how the user's client device interacts with the application. The attributes on the Client Device tab are as follows:

- Keyboard Map: Locked
- Window Management Keys
- Euro Character
- Copy and Paste: Application's Clipboard Security Level
- Audio Redirection Library
- Mouse
- Middle Mouse Timeout
- Monitor Resolution

The *Hosting Application Servers tab* lists the application servers hosting the application. See [Hosting Application Servers Tab](#).

The *Assigned User Profiles tab* lists the user profile objects that can run the application. See [Assigned User Profiles Tab](#).

The *Application Sessions tab* lists the running and suspended application sessions for the application. See [Application Sessions Tab](#).

Attributes Reference

This section describes the available attributes for the SGD objects.

For each attribute, usage information is given for the Administration Console. The corresponding command line is also described, where applicable.

Address

Usage: Type a Domain Name System (DNS) name, or Internet Protocol (IP) address, in the field.

Application server objects have this attribute.

Description

This attribute specifies the network address of the application server.

It is best to use the DNS name.

When you create a new application server object, the Name setting is automatically entered in the Address field.

You can use the Test button to validate that the DNS name or IP address is a valid network address. To enable the Test button, you must first save any changes you make to the General tab.

Command Line

Command option: `--address address`

Usage: Replace *address* with a DNS name, preferably, or an IP address.

The following example specifies the address of the application server as `naples.indigo-insurance.com`.

```
--address naples.indigo-insurance.com
```

Allow SSH Downgrade

Usage: Select or deselect the check box.

X application objects have this attribute.

Description

Defines whether SGD can try to display an X application using a regular X11 connection when SSH is used as the connection method and X11 forwarding is not configured or not working.

Command Line

Command option: `--allowsshdwngrade true|false`

Usage: Specify true or false.

The following example specifies that SGD can try to use an X11 connection if X11 forwarding is not available or working.

```
--allowsshdwngrade true
```

Answerback Message

Usage: Type a text string in the field.

Character application objects have this attribute.

Description

Defines the message to return when an inquiry is sent from the application server to the emulator.

This attribute applies to VT420 and Wyse 60 character applications only.

Command Line

Command option: `--answermsg message`

Usage: Replace *message* with the text string to use.

The following example returns the text “My message” in response to an inquiry from the application server.

```
--answermsg "My message"
```

Application Command

Usage: Type the full path name of the application in the field.

The following objects have this attribute:

- Character application
- Windows application
- X application

Description

This attribute specifies the application that runs when users click the link for the application on the webtop or in the desktop Start or Launch menu.

The path name must be the same on all application servers that might run the application.

For any command-line arguments, use the [Arguments for Command](#) attribute.

With X applications, use the [Window Manager](#) attribute to start a window manager for the application.

With Windows applications, you can use a backslash (\) or a forward slash (/) between subdirectories. On the command line you might need to escape backslashes, for example, \\.

With Windows applications, leave the field blank to start a full Microsoft Windows session rather than a particular application.

Command Line

Command option: `--app pathname`

Usage: Replace *pathname* with the full path name of the application. Make sure that you quote any path names containing spaces

The following example specifies a UNIX® platform X application.

```
--app /usr/local/bin/xfinance
```

The following example specifies a Windows application.

```
--app "c:/Program Files/Indigo Insurance/cash.exe"
```

Application Load Balancing

Usage: Select the Override Global Setting check box, and then select an option. To use the global setting defined in the Global Settings tab, deselect the Override Global Setting box.

The following objects have this attribute:

- Character application
- Windows application
- X application

Description

When the application is started, this setting determines the algorithm SGD uses to choose the application server to run the application. The server is selected from those defined on the application object's [Hosting Application Servers Tab](#).

The default setting for this attribute is to use the setting defined on the Global Settings → Performance tab. You can override this by selecting the Override Global Setting check box and selecting an option.

The Administration Console options and their command line equivalents are shown in the following table.

| Administration Console | Command Line | Description |
|--------------------------------------|--------------|--|
| Override Global Setting (deselected) | default | Use the default algorithm defined on the Global Settings → Performance tab. |
| Most Free Memory | memory | Choose the application server with the most free memory. |
| Least CPU Usage | cpu | Choose the application server with the most central processing unit (CPU) idle time. |
| Fewest Applications | sessions | Choose the application server that is running the fewest application sessions through SGD. |

Note – To use the Least CPU Usage and Most Free Memory algorithms, you must install the SGD Enhancement Module on the application server.

Command Line

Command option: `--loadbal default | cpu | memory | sessions`

Usage: Specify a setting.

The following example uses the application server with the most free memory to run the application.

`--loadbal memory`

Application Resumability

Usage: Select an option.

The following objects have this attribute:

- Character application
- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute determines for how long a user is able to resume an application.

| Administration Console | Command Line | Description |
|-------------------------|--------------|---|
| Never | never | The application can never be resumed. Use for applications that do not provide a mechanism for the user to exit. For example, a clock application. |
| During the User Session | session | The application keeps running and is resumable until the user logs out of SGD. If a user does not log out of SGD cleanly, for example, if they close their web browser or terminate the SGD Client without logging out, then applications that are user session resumable keep running for a time. See Application Resumability: Timeout . This is the default setting. |
| General | always | The application keeps running for a time, see Application Resumability: Timeout , after the user logs out of SGD, and can be resumed when they next log in. Use for applications that need to exit in a controlled way. For example, an email application that might need to remove lock files before it exits. |

An *X application* configured with a [Window Type](#) setting of Local X Server is not resumable, whatever the value of the Application Resumability attribute.

A *Windows application* configured to run on the client device, see [Local Client Launch](#), is not resumable, whatever the value of the Application Resumability attribute.

Users can see if an application is resumable or not by pointing to its link on the webtop and looking at the popup window that is displayed.

The webtop has controls for suspending and resuming individual application sessions. If you are using the SGD Client in Integrated mode, applications that have a General resumability setting are automatically suspended when you log out. When you log in again, they are automatically resumed.

Command Line

Command option: `--resumable never | session | always`

Usage: Specify one of the valid resumability settings.

In the following example, the application is never resumable.

```
--resumable never
```

In the following example, the application is resumable until the user logs out of SGD.

`--resumable session`

Application Resumability: Timeout

Usage: Type the number of minutes you want the application to be resumable for in the field.

The following objects have this attribute:

- Character application
- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute ensures that resources on the SGD server are used as efficiently as possible. It is used with the [Application Resumability](#) attribute to define when the SGD server ends a suspended application session.

| Application Resumability Setting | Resumability Behavior |
|----------------------------------|--|
| Never | Ignored. |
| During the User Session | <p>If the SGD Client connection is lost, a timer starts. Once the timer reaches the value of this timeout, the SGD server ends the application session. If the user logs out of SGD, the application session ends.</p> <p>If an application is terminated because the SGD Client exits unexpectedly, the timeout is the timeout plus 20 minutes.</p> |
| General | <p>If the user disconnects from the SGD server in any way, including by logging out, or if the SGD Client connection is lost, a timer starts. Once the timer reaches the value of this timeout, the SGD server ends the application session.</p> <p>If an application is terminated because the SGD Client exits unexpectedly, the timeout is the timeout plus 20 minutes.</p> |

If you leave this setting blank, the default timeout for the Application Resumability attribute is used. You can configure the default timeouts on the Global Settings → Communication tab of the Administration Console.

Command Line

Command option: `--resumetimeout mins`

Usage: Replace *mins* with the number of minutes you want the application to be resumable for.

The following example configures the application to be resumable for at least 30 minutes. This timeout is appropriate for an application configured to be resumable During the User Session.

```
--resumetimeout 30
```

Application Sessions Tab

Usage: Use the buttons in the Application Sessions tab to view and manage application sessions.

The following objects have this attribute:

- Character application
- Windows application
- X application
- 3270 application
- 5250 application
- User profile
- Application server
- Dynamic application

Description

This tab lists the running and suspended application sessions for the selected object. An application session represents an application running on an application server on behalf of a user.

To show more details about an application session, select the check box for the application session in the Application Session List table and click the View Details button.

To end an application session, select the check box for the application session in the Application Session List table and click the End button.

To shadow an application session, select the check box for the application session in the Application Session List table and click the Shadow button. Suspended applications or character applications cannot be shadowed.

Note – In some countries, it is illegal to shadow a user without their knowledge. It is your responsibility to comply with the law.

The Reload button refreshes the Application Session List table.

You can use the Search options to search the Application Session List table. The number of results returned by a search is limited to 150, by default. When searching for a User Identity, User Profile, Secure Global Desktop Server, or Application Server, you can use the "*" wildcard in your search string. Typing a search string of *name* is equivalent to searching for "**name**" and returns any match of the search string. The Start Time search option enables you to search for application sessions that were started between a range of times that you specify. The format of each time specified is *yyyy/mm/dd hh:mm:ss*. Use one of the following formats to specify the range:

- **starttime - endtime**

For example, 2010/05/11 08:00:00 - 2010/05/11 17:00:00.

Searches for application sessions that were started between the specified times.

- **starttime-**

For example, 2010/05/11 08:00:00-.

Searches for application sessions that were started on or after the specified time.

- **-endtime**

For example, -2010/05/11 08:00:33.

Searches for application sessions that were started before the specified time.

Command Line

On the command line, use the `tarantella emulatorsession` commands to list, end, or shadow application sessions. See [“The tarantella emulatorsession Command” on page 692](#).

Command option: `tarantella emulatorsession list --person pobj`

Usage: Replace *pobj* with the user identity of the user.

The following example lists application sessions for the Indigo Jones user profile object.

```
tarantella emulatorsession list --person \  
"o=Indigo Insurance/ou=IT/cn=Indigo Jones"
```

Application Start

Usage: Select or deselect the check box.

Application server objects have this attribute.

Description

This attribute specifies whether applications can run on this application server.

Selecting the check box allows applications to run. The check box is selected by default. An application is started on the application server only if both of the following are true:

- The application server object appears on the application object's [Hosting Application Servers Tab](#).
- The application's load balancing algorithm chooses this application server.

Deselecting the check box means that no new applications can be started on the application server. Making an application server unavailable does not affect applications that are already running. If a user has a suspended application session on the application server and the application is set up to be always resumable, the user can resume their session.

You can use this attribute, for example, to make an application server temporarily unavailable while you carry out maintenance work. If the application server is the only server configured to run a particular application, then the application is not available to users.

Command Line

Command option: `--available true | false`

Usage: Specify true or false.

The following example enables the application server object to run applications.

```
--available true
```

Arguments

Usage: Type Remote Desktop Client command-line arguments in the field.

Windows application objects have this attribute.

Description

This attribute specifies command-line arguments to use for the SGD Remote Desktop Client (ttatssc).

See [“The SGD Remote Desktop Client” on page 149](#) for details of the supported command-line arguments.

Command Line

Command option: `--protoargs args`

Usage: Replace *args* with the command-line arguments for the RDP protocol.

The following example configures high audio quality for the Windows application.

```
--protoargs "-audioquality high"
```

Arguments for Command

Usage: Type the command-line arguments for the application in the field.

The following objects have this attribute:

- Character application
- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute specifies the command-line arguments to use when starting the application. The [Application Command](#) attribute specifies the application that runs, without arguments.

For X applications, *do not* include the `-display` argument. The display is set automatically for each user.

Command Line

Command option: `--args args`

Usage: Replace *args* with the command-line arguments for the application. Make sure you quote the arguments.

The following example runs the application with command-line arguments to set the background color to “plum4”.

```
--args "-bg plum4"
```

Assigned Applications Tab

Usage: To assign applications to a user profile, organization, or organizational unit object, click the Add button in the Editable Assignments table.

To delete applications for a user profile, organization, or organizational unit object, use the Delete button in the Editable Assignments table.

The following objects have this attribute:

- Organization
- Organizational Unit
- User profile

Description

The Assigned Applications tab lists the applications that are assigned to the selected user profile, organizational unit, or organization.

This attribute defines a series of application links available to the user. Each link is stored as *a reference to the application object*, so the same application object can be assigned to many users. If an object is moved or renamed later, all references to the object are automatically updated.

If a group of applications is added to an Assigned Applications tab, the group’s members and not the group are assigned.

User profile objects and organizational unit objects can inherit applications from their parent in the organizational hierarchy. See [Inherit Assigned Applications from Parent](#). To inherit applications assigned to the parent object, select the Inherit Assigned Applications from Parent check box in the Editable Assignments area.

The following sections of the Assigned Applications tab are used to display, select, and assign applications:

- Effective Applications table
- Editable Assignments table

Effective Applications Table

The Effective Applications table shows all the application objects that are assigned to the selected object. The Local Assignments section of the table lists applications that are selected from the local repository.

The Assignment Type column shows one of the following:

- **Direct.** The assignment was made using the Editable Assignments table.
- **Indirect.** The assignment is the result of another relationship, such as membership of a group, or inheritance from another object.
- **Multiple.** The assignment has multiple sources, both Direct and Indirect.

If an assignment type is Indirect or Multiple, clicking the See Details link displays information that enables you to trace the origin of the link.

Editable Assignments Table

You can use the Editable Assignments table to select applications from the local repository.

Click the Add button in the Editable Assignments table. The Add Application Assignment window is shown.

To select applications in the Add Application Assignment window, do either of the following:

- **Browse the Navigation Tree.** As you browse the tree, the Content Area is updated with applications.
- **Use the Search Applications field.** Use this field to search for applications. Type in the names of applications in the field. Note that you can use the "*" wildcard in your search string. Typing a search string of *name* is equivalent to searching for "**name**" and returns any match of the search string. Results of the search are displayed in the Search Results table in the Content Area. The number of results returned by a search is limited to 150, by default.

Select the required applications from those listed in the Content Area. When you have finished selecting applications click the Add button.

The selected applications are displayed in the Effective Applications table of the Assigned Applications tab.

To delete applications from the Assigned Applications tab, use the Delete button in the Editable Assignments table.

Command Line

Command option: `--links object`

Usage: Replace *object* with the full name of the object. For example, "o=applications/ou=Finance/cn=XClaim". Make sure that you quote any object names containing spaces.

The following example adds Pers-o-dat and Slide-o-win as links on a webtop.

```
--links "o=applications/cn=Pers-o-dat" \
        "o=applications/cn=Slide-o-win"
```

Assigned User Profiles Tab

Usage: To assign user profiles to an application, click the Add button in the Editable Assignments table. If you are using SGD with a Lightweight Directory Access Protocol (LDAP) directory, you can also use the LDAP Searches area of the Assigned User Profiles tab to search for users in your LDAP directory server.

The following objects have this attribute:

- Character application
- Document
- Group
- Windows application
- X application
- 3270 application
- 5250 application
- Dynamic application

Description

Use this tab to define the user profile objects that can run an application, or group of applications. The application, or group of applications, is *in addition* to any applications already defined for the user profile in its Assigned Applications tab.

User profile objects can be selected from the local repository. If you are using an LDAP directory, you can also select the following:

- Users in the LDAP directory
- Groups of users in the LDAP directory
- Users in the LDAP directory that match an LDAP search criteria

The following sections of the Assigned User Profiles tab are used to display, select and assign user profile objects:

- Effective User Profiles table
- Editable Assignments table
- LDAP Searches section

Effective User Profiles Table

The Effective User Profiles table shows all the user profile objects that are assigned to the application.

The Local Assignments section of the table lists user profiles that are selected from the local repository.

The LDAP Assignments section of the table lists users and groups that are selected from an LDAP directory. This section is only shown if the Local + LDAP setting is selected for the Repository field in the User Profiles tab. You can click the Load LDAP Assignments link to refresh this area of the table.

The Assignment Type column shows one of the following:

- **Direct.** The assignment was made using the Editable Assignments table.
- **Indirect.** The assignment is the result of another relationship, such as an LDAP search, membership of a group, or inheritance from another object.
- **Multiple.** The assignment has multiple sources, both Direct and Indirect.

If an assignment type is Indirect or Multiple, clicking the See Details link displays information that enables you to trace the origin of the link.

Editable Assignments Table

You can use the Editable Assignments table to select user profile objects from the local repository, and, if you are using LDAP authentication, users, or groups in an LDAP directory.

Click the Add button in the Editable Assignments table. The Add User Assignment window is shown.

The Add User Assignment window can be used to select the following:

- User profiles from the local repository
- Users in an LDAP directory
- Groups in an LDAP directory

To use the local repository, select the Local option in the Repository list.

To use the local repository *and* your LDAP directory server, select the Local + LDAP option in the Repository list.

To select user profiles in the Add User Assignment window, do either of the following:

- **Browse the Navigation Tree.** As you browse the tree, the Content Area is updated with user profiles.
- **Use the Search User Profiles field.** Use this field to search the user profiles within the selected repositories. You can type in names of users and groups in your LDAP directory. Note that you can use the "*" wildcard in your search string. Typing a search string of *name* is equivalent to searching for "**name**" and returns any match of the search string. Results of the search are displayed in the Search Results table in the Content Area. The number of results returned by a search is limited to 150, by default. The Matched Attribute field of the Search Results table indicates the LDAP attribute that the search matched on.

Select the required user profiles from those listed in the Content Area. When you have finished selecting user profiles, click the Add button.

The selected user profiles are displayed in the Effective User Profiles table of the Assigned User Profiles tab.

To delete applications you have added to the Assigned User Profiles tab, use the Delete button in the Editable Assignments table.

LDAP Searches Section

The LDAP Searches section is used to define search criteria for locating users in an LDAP directory. You can use this feature to assign an application or group of applications to all users in an LDAP directory that match the search criteria.

The search criteria can be either of the following:

- An RFC2254-compliant LDAP search filter
- An RFC1959-compliant LDAP URL

For an RFC2254 search filter, enclose each search criteria in double quotes and brackets.

For an LDAP URL, use the format `ldap://search-criteria`. If you include the host, port and return attribute specification in the URL they are ignored. This is because the LDAP directory server configured as part of SGD authentication is used.

The LDAP Search area includes two options:

- **Simple Search.** This option enables an area where you can "build" a simple LDAP search filter using the window controls. In the Filter Components table, select the attributes you want to match and define search criteria for them.

- **Advanced Search.** This option displays a field where you can type in an LDAP URL or search filter.

The Simple Search option is designed for creating LDAP search filters that are based on attributes such as `cn` and `uid`. The Advanced Search option enables you to create more complex LDAP search filters.

As you build a simple search, the LDAP filter string is shown in gray text in the Advanced Search area. If you then select the Advanced Search option, the LDAP filter string can be edited. This enables you to start with a simple search and then edit the search string manually to specify an advanced search.

You cannot revert to a simple search after specifying an advanced search that is incompatible with the capabilities of the simple search. You must delete the advanced search and re-enter the simple search.

To specify where in the LDAP directory to start searching from, click the Browse button next to the Search Root field. You can then use the Select Root for LDAP Search window to browse or search for a location in the LDAP directory. Selecting a new Search Root loads a new LDAP URL. The new URL is indicated next to the Browse button and in the Advanced Search box.

Select the Search Filter options to specify the attributes you want to match in your search. You can choose to match all of the attributes (Match All), any of the attributes (Match Any), or none of the attributes (Match None).

Note – The Administration Console does not automatically escape the special characters specified in [RFC2254](#). To use a special character in the Administration Console, you must manually type the escape sequence. For example, to search for a user with the common name “John Doe (123456)”, type the following `cn=John Doe\0x28123456\0x29` in the search field. The command line does escape the special characters.

SGD supports the use of extensible matching search filters as specified in [RFC2254](#). This enables you to look up information from components that make up an object’s distinguished name (DN). For example, to assign an application to a user that is contained within any OU called managers (`ou=managers`), you can use a `(&(ou:dn:=managers))` search filter.

As you configure LDAP searches, use the Preview button to check that the search returns the expected results.

To save the LDAP search definition, click the Save button.

Click the Load LDAP Assignments link in the Effective User Profiles tab. The user profiles from the LDAP search are displayed in the LDAP Assignments section of the Effective User Profiles table.

Command Line

On the command line, make sure that you quote any object names containing spaces.

LDAP Users

Command option: `--ldapusers user_dn`

Usage: Enter one or more DN's of users in an LDAP directory.

The following example assigns the application or groups of applications to users with the UID "violet" in the Sales department and the UID "emmarald" in the Marketing department.

```
--ldapusers uid=violet,ou=Sales,dc=indigo-insurance,dc=com \
uid=emmarald,ou=Marketing,dc=indigo-insurance,dc=com
```

LDAP Groups

Command option: `--ldapgroups group_dn`

Usage: Enter one or more DN's of groups in an LDAP directory.

If your organization uses nested groups (sub-groups), you might need to change the depth of the group search.

The following example assigns the application or groups of applications to users in the managers group in the Sales and Marketing departments.

```
--ldapgroups cn=managers,ou=Sales,dc=indigo-insurance,dc=com cn=
managers,ou=Marketing,dc=indigo-insurance,dc=com
```

LDAP Search

Command option: `--ldapsearch search_string`

Usage: Enter one or more LDAP search strings.

The following example assigns the application or groups of applications to any manager in the Sales department *and* anyone who has Violet Carson as their manager.

```
--ldapsearch "(&(job=manager)(dept=Sales))" \
"(manager=Violet Carson)"
```

The following example assigns the application or groups of applications to any manager in the Sales department of indigo-insurance.com.

```
--ldapsearch "ldap:///ou=Sales,dc=indigo-insurance,dc=com??sub?job=manager"
```

Attribute Map

Usage: Type the full path name of the attribute map in the field.

Character application objects have this attribute.

Description

This attribute specifies the attribute map to use for the application. This maps attributes such as bold and underline to colors.

To use the default attribute map, leave the setting blank.

An example attribute map is installed in
`/opt/tarantella/etc/data/attrmap.txt`.

Command Line

Command option: `--attributemap attrmap`

Usage: Replace *attrmap* with the full path name of the attribute map to use.

The following example uses the named attribute map.

```
--attributemap /opt/tarantella/etc/data/myattrmap.txt
```

Audio Redirection Library

Usage: Select or deselect the check box.

X application objects have this attribute.

Description

This attribute specifies whether the application enables the SGD audio redirection library.

Some X applications are hard-coded to use the `/dev/audio` or `/dev/dsp` devices for audio output. Enabling the audio redirection library causes the application to use the device specified by the `SGDAUDIODEV` environment variable instead.

Command Line

Command option: `--unixaudiopreload true | false`

Usage: Specify `true` or `false`.

The following example enables the audio redirection library for the application.

```
--unixaudiopreload true
```

Background Color

Usage: Type a valid color resource, such as `yellow`, in the field.

The following objects have this attribute:

- 3270 application
- 5250 application

Description

Specifies the background color of the application's text window.

Color names are resolved to RGB values using the file named in the X Protocol Engine's [RGB Database](#) attribute.

Command Line

Command option: `--3270bg color`

Command option: `--bg color`

Usage: Replace *color* with a valid color resource, such as `yellow`.

In the following example, the background color of the 3270 application text window is set to the color `plum4`.

```
--3270bg plum4
```

In the following example, the background color of the 5250 application text window is set to the color plum4.

```
--bg plum4
```

Bandwidth Limit

Usage: Select the maximum bandwidth from the list.

User profile objects have this attribute.

Description

This attribute specifies the maximum bandwidth a user can use between the client device and the SGD server for X and Windows applications.

Select None to specify no limit. The user can then use as much of the available bandwidth as possible. This gives the best application usability for the speed of the network connection.

You do not need to change this unless you have particular bandwidth restrictions. For normal use, use None.

The table below shows the bandwidth settings in the Administration Console and the equivalent values to use on the command line:

| Administration Console | Command Line |
|------------------------|--------------|
| 2400 bps | 2400 |
| 4800 bps | 4800 |
| 9600 bps | 9600 |
| 14.4 Kbps | 14400 |
| 19.2 Kbps | 19200 |
| 28.8 Kbps | 28800 |
| 33.6 Kbps | 33600 |
| 38.8 Kbps | 38800 |
| 57.6 Kbps | 57600 |
| 64 Kbps | 64000 |
| 128 Kbps | 128000 |
| 256 Kbps | 256000 |

| Administration Console | Command Line |
|------------------------|--------------|
| 512 Kbps | 512000 |
| 768 Kbps | 768000 |
| 1 Mbps | 1000000 |
| 1.5 Mbps | 1500000 |
| 10 Mbps | 10000000 |
| None | 0 |

Command Line

Command option: `--bandwidth bandwidth`

Usage: Replace *bandwidth* with the maximum bandwidth, in bits per second.

The following example limits the user to a maximum bandwidth of 512 kilobits per second.

```
--bandwidth 512000
```

The following example enables the user to use as much of the available bandwidth as possible.

```
--bandwidth 0
```

Border Style

Usage: Select an option.

Character application objects have this attribute.

Description

This attribute determines whether the terminal window has a raised, indented, or “flat” (normal) appearance.

Command Line

Command option: `--border normal | indented | raised`

Usage: Specify the border style you want.

In the following example, the terminal window has a raised appearance.

```
--border raised
```

Client Drive Mapping

Usage: Use the Client Drive Mapping table to create client drive mapping (CDM) specifications. Use the Add, Edit and Delete buttons to create, edit and remove CDM specifications. Order the specifications using the Move Up and Move Down buttons. Any CDM specifications you create are listed in the Mappings Defined Directly section of the Client Drive Mapping table.

The following objects have this attribute:

- Organization
- Organizational Unit
- User profile
- Windows application

Description

This attribute defines the drives on their client device that a user can access from applications running on Microsoft Windows, UNIX, and Linux platform application servers.

Using this attribute, you configure the client drives you want users to access for user profiles, organizational unit, and organization objects. CDM uses inheritance. You define access to client drives at an organization level, which you can override at an organizational unit level or user profile level. By default, users have read and write access to all drives.

For Windows applications, you can configure application-specific client drive access with this attribute.

For Windows application objects, this setting overrides the CDM configuration for organization, organizational unit, or user profile objects. The order of precedence for CDM configuration is: Windows application → user profile → organizational unit → organization.

The Client Drive Mapping attribute is an *ordered list* of drive mapping specifications. Each specification includes the following:

- The client drive letter or type

- The access rights to grant to the client drive

Note – The first matching entry in the list is used, so make sure that the most specific settings, for example A or B, appear before more general settings, for example All Drives.

The following tables show the available options for each part of a drive mapping specification, and the corresponding value to use on the command line.

The following Client Device Drive options are available.

| Administration Console | Command Line |
|------------------------|---------------|
| All Drives | alldrives |
| Fixed Drives | fixeddrives |
| R/W Removable | rw |
| R/O Removable | ro |
| Network Drives | networkdrives |
| Removable Devices | removable |
| A:, B: ... Z: | a, b ... z |

Note – Client Device Drive options that specify a drive letter, such as A:, are only supported for Microsoft Windows client devices.

The following Access Rights options are available.

| Administration Console | Command Line |
|------------------------|--------------|
| Read Only | ro |
| Read/Write | rw |
| None | none |

Command Line

Command option: `--cdm drive_spec`

Usage: Replace *drive_spec* with a drive mapping specification of the form *clientdrive:access*. For example, *a:rw*. Separate each *drive_spec* with the pipe character, (|).

For a user profile object, the following example means the user is given read-write access to drive A on their client device, and also has read-write access to all network drives defined on their client device. The same drive letter is used on the client device.

```
--cdm 'a:rw|networkdrives:rw'
```

The user might have access to other drives, for example a fixed drive C, depending on the Client Drive Mapping attributes for the user profile object's ancestors in the organizational hierarchy, or any Client Drive Mapping attributes configured for a specific Windows application object.

Client Printing

Usage: Select an option.

The following objects have this attribute:

- Organization
- Organizational Unit
- User profile
- Windows application

Description

Controls the client printers that users can print to when printing from Windows applications.

This attribute can only be edited using the Administration Console if [Client Printing: Override](#) is enabled for the object.

The setting for this attribute overrides the following:

- The setting for a parent object in the organizational hierarchy.
- The default setting configured on the Global Settings → Printing tab of the Administration Console, if no parent object configuration exists.
- For Windows application objects, this setting overrides the printing configuration for organization, organizational unit, or user profile objects. The order of precedence for printing configuration is: Windows application → user profile → organizational unit → organization.

For organization, organizational unit, and user profile objects, changes to this attribute only take effect for new user sessions.

If you select No Printer, you can still use an SGD Portable Document Format (PDF) printer.

The Administration Console options and their command line equivalents are shown in the following table.

| Administration Console | Command Line | Description |
|------------------------|--------------|---|
| All Printers | 2 | Let users print to all client printers |
| Default Printer | 1 | Let users print to client's default printer |
| No Printer | 0 | No client printers are available |

If users can only print to their default printer and they want to print to a different printer, they have to log out of SGD, change the default printer and then log in again.

Command Line

Command option: `--mapprinters 2|1|0`

Usage: Specify 2|1|0.

The following example enables users to print only to their default client printer.

```
--mapprinters 1
```

Client Printing: Override

Usage: For *user profile* objects or *organizational unit* objects, select the Override Parent's Settings check box. To use the setting defined for the parent object, deselect the Override Parent's Settings check box.

For *organization* or *Windows application* objects, select the Override Global Settings check box. To use the default setting defined in the Global Settings → Client Device tab, deselect the Override Global Settings check box.

The following objects have this attribute:

- Organization
- Organizational Unit
- User profile
- Windows application

Description

Enables *user-specific* or *application-specific* printing configuration. This configuration is used when printing from Windows applications.

For user-specific printing configuration, client printing override is enabled for a *user profile*, *organizational unit*, or *organization* object. In this case, the printing settings for the object override the following:

- The printing settings for a parent object in the organizational hierarchy.
- The default printing settings configured on the Global Settings → Printing tab of the Administration Console, if no parent object printing configuration exists.

For application-specific printing configuration, client printing override is enabled for a *Windows application* object. In this case, the printing settings for the object override the following:

- The printing settings for organization, organizational unit, or user profile objects. The order of precedence for printing configuration is: Windows application → user profile → organizational unit → organization.
- The default printing settings configured on the Global Settings → Printing tab of the Administration Console.

For organization, organizational unit, and user profile objects, changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--userprintingconfig 1|0`

Command option: `--appprintingconfig 1|0`

Usage: Specify 1 (true) or 0 (false).

The following example enables user-specific printing configuration.

```
--userprintingconfig 1
```

The following example enables application-specific printing configuration for a Windows application object.

```
--appprintingconfig 1
```

Client Profile Editing

Usage: For *user profile* objects or *organizational unit* objects, select the Override Parent's Setting check box and then select or deselect the Enabled option. To use the setting defined for the parent object, deselect the Override Parent's Setting check box.

For *organization* objects, select the Override Global Setting check box and then select or deselect the Enabled option. To use the default setting defined in the Global Settings tab, deselect the Override Global Setting check box.

The following objects have this attribute:

- Organization
- Organizational Unit
- User profile

Description

This attribute controls whether or not users can create and edit profiles for use with the SGD Client.

Note – Profile editing must also be enabled on the Global Settings → Client Device tab of the Administration Console.

The Administration Console options and their command line equivalents are shown in the following table.

| Administration Console | Command Line | Description |
|--|--------------|---|
| Override Parent's Setting (deselected) | 2 | User profile objects or organizational unit objects. Use the setting inherited from the parent object. This is the default setting. |
| Override Global Setting (deselected) | 2 | Organization objects. Use the global setting. This is the default setting. |
| Enabled (selected) | 1 | Enable client profile editing. |
| Enabled (deselected) | 0 | Disable client profile editing. |

For user profile objects or organizational unit objects, deselect the Override Parent's Setting check box to inherit the setting of a parent object in the organizational hierarchy. This is used to enable or disable profile editing for many users without having to edit each user profile object.

For organization objects, deselect the Override Global Setting check box to use the default setting configured on the Global Settings → Client Device tab of the Administration Console.

SGD checks the user profile object for the user and then any parent object further up the organizational hierarchy to see whether profile editing is enabled or disabled. If all the objects selected are configured to use the parent's setting, then the default setting is used.

If profile editing is disabled for a user profile object in the System Objects organization, for example `o=Tarantella System Objects/cn=UNIX User Profile`, this affects *all* users who are assigned that profile.

By default, profile editing is enabled.

Command Line

Command option: `--editprofile 2|1|0`

Usage: Specify 2|1|0.

The following example disables profile editing.

```
--editprofile 0
```

Code Page

Usage: Select an option.

Character application objects have this attribute.

Description

This attribute specifies the code page you want to use for the emulator. Different code pages are available for different types of character application.

| Application Type | Code Pages Available |
|------------------|---|
| SCO Console | <ul style="list-style-type: none">• 437 - International• 850 - Multilingual• 852 - Central Europe• 860 - Portuguese• 863 - Canadian-French• 865 - Danish-Norwegian |
| VT420 | <ul style="list-style-type: none">• 8859-1 - ISO Latin 1• 8859-2 - ISO Latin 2 |
| Wyse 60 | <ul style="list-style-type: none">• Multinational• Mazovia• CP852 |

Command Line

Command option: `--codepage 437 | 850 | 852 | 860 | 863 | 865 | 8859-1 | 8859-2 | Multinational | Mazovia | CP852`

Usage: Specify a valid setting for the type of character application.

The following example uses the ISO 8859-1 code page, appropriate for a VT420 application.

```
--codepage 8859-1
```

Color Depth

Usage: Select a setting from the list.

The following objects have this attribute:

- X application
- Windows application

Description

The color depth for the application. As the number of colors increases, more memory is required on the SGD server and on the client device, and more network bandwidth is used between them.

X Applications

The 16/8-bit, 24/8-bit, 8/16-bit, and 8/24-bit settings enable you to support X applications with multiple color depths. For example, if you need to run an 8-bit application in a 16-bit or 24-bit high color X application session, such as a Common Desktop Environment (CDE) desktop, use either the 16/8-bit or the 24/8-bit setting.

Changing these settings can affect system performance as follows:

- Increases the amount of memory used on the SGD server compared to an application using a single color depth.

The amount of extra memory used for each setting is as follows:

- The 8/16 setting uses 200% more memory
- The 8/24 setting uses 400% more memory
- The 16/8 setting uses 50% more memory
- The 24/8 setting uses 25% more memory
- Increases the amount of bandwidth used.
- Degrades performance over low bandwidth connections.

To reduce network bandwidth at greater color depths for X applications, change the [Color Quality](#) setting.

Windows Applications

SGD supports 8-bit, 16-bit, 24-bit, and 32-bit color depths in a Windows Terminal Server session. 32-bit color depths are supported for Windows Server 2008 (including Windows Vista) and later versions.

If the application server is configured to use a lower color depth than the color depth configured for the application object, SGD automatically adjusts the color depth to match the server setting.

The 24/32-bit setting means that the color depth is 32-bit on platforms that support 32-bit color. For other platforms, the application is displayed using 24-bit color.

Command Line

Command option: --depth 8 | 16 | 24 | 16/8 | 24/8 | 8/16 | 8/24 | 24/32

Usage: Specify a valid setting.

The 16/8-bit, 24/8-bit, 8/16-bit, and 8/24-bit settings apply only to X applications.

The 24/32-bit setting applies only to Windows applications.

The following example sets the color depth for the application to 16-bit color (thousands of colors).

```
--depth 16
```

Color Map

Usage: Type the full path name of the color map in the field.

Character application objects have this attribute.

Description

This attribute specifies the color map to use for the application. A color map maps logical colors such as Color_1, Color_2 and so on, to displayed colors.

To use the default color map, /opt/tarantella/etc/data/colormap.txt, leave the setting blank.

Command Line

Command option: --colormap *colormap*

Usage: Replace *colormap* with the full path name of the color map to use.

The following example uses the named color map.

```
--colormap /usr/local/maps/mycolormap.txt
```

Color Quality

Usage: Select a setting from the list.

X application objects have this attribute.

Description

The effective color depth displayed on client devices. Reducing color quality reduces bandwidth usage, but also reduces the number of colors that can be displayed.

Note – If the [Color Depth](#) is set to 8-bit, this attribute is not available. If the Color Depth is set to 16-bit, only the 16-bit, 15-bit, 12-bit, 9-bit, and 6-bit settings are available.

The default setting Best at Application Start fixes the color depth at the most appropriate setting according to network conditions at the time the user starts the application. The color depth does not change while the session is running.

Specify Adjust Dynamically to enable the quality level to change at any time during the session, depending on network conditions. This setting works within the following ranges:

- **24 bit images** – 12 to 24-bit color
- **16 bit images** – 12 to 16-bit color

The following table shows the effect on color quality of using a numeric quality setting.

| Color Quality Setting | Approximate Color Quality for 16-bit Applications | Approximate Color Quality for 24-bit Applications |
|-----------------------|---|---|
| 24 | - | 100% |
| 21 | - | 88% |
| 18 | - | 75% |
| 16 | 100% | 67% |
| 15 | 94% | 63% |
| 12 | 75% | 50% |
| 9 | 56% | 38% |
| 6 | 38% | 25% |

The physical color quality of the client device is not forced to match that of the X session. If a 24-bit color session is being displayed on an 8-bit client device, the client dithers the image locally so that the session can be displayed reasonably.

Command Line

Command option: `--quality automatic|best|24|21|18|16|15|12|9|6`

Usage: Specify a valid setting.

The following example sets the color quality to 12-bit color. If the [Color Depth](#) is set to 24-bit, this reduces color quality to approximately 50% on client devices.

```
--quality 12
```

Command Compression

Usage: Select an option.

The following objects have this attribute:

- Windows application
- X application
- 3270 application
- 5250 application
- Character application

Description

This attribute determines whether the Adaptive Internet Protocol (AIP) compresses commands for transmission.

Select **Adjust Dynamically** to allow compression to be turned on or off at any stage, according to the network conditions.

With some applications, compression incurs a greater overhead than transmitting commands uncompressed. Turn off compression for these applications.

Command Line

Command option: `--compression automatic|on|off`

Usage: Specify a valid option.

The following example disables AIP command compression.

```
--compression off
```

Command Execution

Usage: Select an option.

The following objects have this attribute:

- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute determines whether the AIP protocol always executes commands in order, or optimizes commands for performance reasons.

Select Adjust Dynamically to allow the network conditions to determine the setting.

For some applications, for example those that use animation, the order that commands are executed is critical.

Command Line

Command option: `--execution automatic|inorder|optimized`

Usage: Specify a valid option. When listing object attributes on the command line, the following applies:

- The `inorder` attribute value is displayed as `on`
- The `optimized` attribute value is displayed as `off`

The following example executes commands in the order they occur.

```
--execution inorder
```

Comment

Usage: Type a description of the object in the field.

The following objects have this attribute:

- Character application
- Document

- Group
- Application server
- Organization
- Organizational Unit
- User profile
- Windows application
- X application
- 3270 application
- 5250 application
- Dynamic application
- Dynamic application server

Description

This attribute describes the object. Use this as an optional comment field for administrator notes

Descriptions can include any characters you want.

Command Line

Command option: `--description text`

Usage: Replace *text* with a description of the object. Ensure that you quote any descriptions containing spaces.

The following example describes the object. You might use this description with a document object, for example.

```
--description "The intranet for Indigo Insurance"
```

Connection Closed Action

Usage: Select a telnet close option.

The following objects have this attribute:

- 3270 application
- 5250 application

Description

Specifies the course of action to be taken by the TeemTalk for Unix emulator when the telnet connection to the application server is closed.

The Administration Console options and their command line equivalents are shown in the following table.

| Administration Console | Command Line | Description |
|------------------------|--------------|---|
| Prompt User for Action | 0 | Prompt the user to choose to either reconnect, close the connection or exit the emulator. |
| Reconnect | 2 | Attempt to reconnect to the 3270 application server. |
| Close Connection | 3 | Close the connection. |
| Exit Emulator | 1 | Exit the TeemTalk for Unix emulator. The SGD application session is terminated. |

Command Line

Command option: `--3270tn 0|1|2|3`

Command option: `--tn 0|1|2|3`

Usage: Specify one of the valid telnet close options.

The following example exits the emulator when the telnet connection to the 3270 application server is closed.

```
--3270tn 1
```

The following example exits the emulator when the telnet connection to the 5250 application server is closed.

```
--tn 1
```

Connection Method

Usage: Select a connection method option.

The following objects have this attribute:

- Character application
- X application

- 3270 application
- 5250 application

Description

This attribute specifies the mechanism used by the SGD server to access the application server and start the application.

The default connection method is `telnet`.

For character applications, only the connection methods `telnet` and `ssh` are allowed.

Command Line

Command option: `--method rexec | telnet | ssh`

Usage: Specify one of the valid connection methods. Not all methods are available for all types of application.

The following example uses the `telnet` connection method to log in to an application server.

```
--method telnet
```

Connections

Usage: Create as many connection type specifications as you need, using the Connection Definitions table. Use the Add, Edit, and Delete buttons to create, modify, and delete connections. Order the connections using the Move Up and Move Down buttons.

The following objects have this attribute:

- Organization
- Organizational Unit
- User profile

Description

This attribute defines, for ranges of DNS names or IP addresses, the connections that are allowed between the client device and the SGD server.

Once a user is logged in to an SGD server, the DNS names and IP addresses of the client device and the SGD server are used to determine the type of connection. First, the Connections attribute for the user profile object is selected. If no matching entry exists, the parent organizational unit's Connections attribute is selected, and so on up the organizational hierarchy to the organization object.

If no matching entry for the organization object is found, the user is given the best available connection.

Processing of connection types is turned off by default, enabling users to log in more quickly. You can turn on processing of connection types on the Security tab in the Global Settings → Security tab of the Administration Console.

The Connections attribute is an *ordered list* of connection type specifications. Each specification names the following:

- The DNS name or IP address of a client device. Use the wildcards ? and * to match more than one client device.
- The DNS name or IP address of an SGD server. Use the wildcards ? and * to match more than one SGD server.
- The connection type.

In all cases, DNS names or IP addresses are considered *from the perspective of the SGD server*. They are peer DNS names and IP addresses. If your network is configured to use different names on each side of a firewall, you must use the names on the side of the SGD servers for this attribute.

The following connection types are available.

| Administration Console | Command Line | Notes |
|------------------------|--------------|--|
| Standard | STD | Always available. |
| Secure | SSL | Gives users a secure connection between their client device and the SGD server. The connection uses Secure Sockets Layer (SSL). Only available if SGD security services are enabled. If not, users configured to receive secure connections are given standard connections instead. |

Note – If security services have been enabled on the SGD server, all connections are secure until the user logs in. Once the user is known, the connection can be downgraded.

Command Line

Command option: `--conntype type_spec`

Usage: Replace *type_spec* with a connection type specification of the form: *client:server:type*. For example, `192.168.5.*:*:STD`.

Separate each *type_spec* with the “pipe” character, “|”.

The following example, for a user profile object, means the user is given a secure connection to all SGD servers if the client device has an IP address that starts 192.168.5, and a standard connection for all other client devices.

```
--conntype '192.168.5.*:*:SSL|*:*:STD'
```

For an organizational unit or an organization object, these connection type specifications are used only if no match is found for the client device and SGD server in the user profile object’s [Connections](#) attribute.

Connection Method: SSH Arguments

Usage: Select the ssh Connection Method option and type the ssh command-line arguments in the field.

The following objects have this attribute:

- Character application
- X application
- 3270 application
- 5250 application

Description

The attribute enables you to specify the command-line arguments for the ssh client when the [Connection Method](#) for an application is ssh.

See [“Using SSH” on page 178](#) for information on using ssh with SGD.

Command Line

Command option: `--ssharguments args`

Usage: Replace *args* with the ssh command-line arguments.

The following example configures the `ssh` client to use the `-X` command-line option when using the application. This enables X11 forwarding.

```
--ssharguments "-X"
```

Console Mode

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

Enabling this attribute starts the Windows application in console mode.

For Windows 2003 versions, the application connects to a user console session on the Windows application server. This is equivalent to using the `/console` option of the `mtsc.exe` (Microsoft Terminal Services Client) program.

For Windows 2008 and later versions, the application connects to an administrator mode console session on the Windows application server. This is equivalent to using the `/admin` option of the `mtsc.exe` program.

Command Line

Command option: `--console 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables console mode.

```
--console 1
```

Copy and Paste

Usage: For *user profile* objects or *organizational unit* objects, select the Override Parent's Setting check box and then select or deselect the Enabled option. To use the setting defined for the parent object, deselect the Override Parent's Setting check box.

For *organization* objects, select the Override Global Setting check box and then select or deselect the Enabled option. To use the default setting defined in the Global Settings → Client Device tab, deselect the Override Global Setting check box.

The following objects have this attribute:

- Organization
- Organizational Unit
- User profile

Description

This attribute controls whether users can use copy and paste in Windows or X application sessions.

For user profile objects or organizational unit objects, deselect the Override Parent's Setting check box to inherit the setting of a parent object in the organizational hierarchy. This is used to enable or disable copy and paste for many users without having to edit each user profile object.

For organization objects, deselect the Override Global Setting check box to use the default setting configured on the Global Settings → Client Device tab of the Administration Console.

When a user starts an application, SGD checks the user profile object for the user and then any parent object further up the organizational hierarchy to see whether copy and paste is enabled or disabled. If all the objects selected are configured to use the parent's setting, then the default setting is used.

By default, copy and paste is enabled.

The Administration Console options and their command line equivalents are shown in the following table.

| Administration Console | Command Line | Description |
|--|--------------|---|
| Override Parent's Setting (deselected) | 2 | User profile objects or organizational unit objects. Use the setting inherited from the parent object. This is the default setting. |
| Override Global Setting (deselected) | 2 | Organization objects. Use the global setting. This is the default setting. |
| Enabled (selected) | 1 | Enable copy and paste. |
| Enabled (deselected) | 0 | Disable copy and paste. |

Changes to this attribute only take effect for new application sessions.

Command Line

Command option: `--clipboard 2|1|0`

Usage: Specify 2|1|0.

The following example disables copy and paste for a user's Windows or X application sessions.

```
--clipboard 0
```

Copy and Paste: Application's Clipboard Security Level

Usage: Select the Enabled check box and type a number in the field.

The following objects have this attribute:

- Windows application
- X application

Description

This attribute is used to control user copy and paste operations in Windows or X application sessions.

Use this attribute to specify a security level. The security level can be any positive integer. The higher the number, the higher the security level.

You can only copy and paste data to an application if the application has the same security level or higher as the source application. The source application is the application the data was copied from.

SGD Clients also have a security level. You can only copy and paste data to applications running on the client device if the client has the same security level or higher as the source application. See [“Client's Clipboard Security Level” on page 472](#).

The default security level is 3.

Changes to this attribute only take effect for new application sessions.

Command Line

Command option: `--clipboardlevel level`

Usage: Replace *level* with the security level. Specify -1 to disable copy and paste operations for the application object.

The following example sets the security level for an application to 5. You can only copy and paste data to this application if the source application or SGD Client has a security level of 5 or less.

```
--clipboardlevel 5
```

Cursor

Usage: Select a cursor style option.

Character application objects have this attribute.

Description

This attribute specifies how you want the cursor to appear within the application.

Command Line

Command option: --cursor off | block | underline

Usage: Specify the cursor style you want.

The following example uses an underline for the cursor.

```
--cursor underline
```

Cursor Key Codes Modification

Usage: Select or deselect the check box.

Character application objects have this attribute.

Description

This attribute specifies the behavior of the cursor keys. It determines whether they always generate cursor movement codes, or whether the application changes the codes generated by the cursor keys.

This attribute applies to VT420 character applications only.

Command Line

Command option: `--cursorkeys application | cursor`

Usage: Specify the cursor key behavior you want.

In the following example, the cursor keys always generate cursor movement codes.

```
--cursorkeys cursor
```

Cursor Settings

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

This attribute specifies whether to enable mouse pointer schemes and customizations for the Windows application. Disabling these features can improve performance.

Command Line

Command option: `--disablecursorsettings 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables mouse pointer schemes and customizations.

```
--disablecursorsettings 1
```

Cursor Shadow

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

This attribute specifies whether to show the mouse pointer shadow for the Windows application. Disabling the mouse pointer shadow can improve performance.

Command Line

Command option: `--disablecursorshadow 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables the mouse pointer shadow.

```
--disablecursorshadow 1
```

Delayed Updates

Usage: Select or deselect the check box.

The following objects have this attribute:

- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute specifies whether delayed updates of the display are enabled. This accumulates changes and can improve performance.

If your application's display must always be exact, deselect the check box. To improve performance, turn off delayed updates for animation.

Command Line

Command option: `--delayed true|false`

Usage: Specify true or false.

The following example enables delayed updates of the application's display.

```
--delayed true
```

Desktop Wallpaper

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

This attribute specifies whether to enable or disable the wallpaper for a Windows application, such as a Windows desktop session. Disabling the wallpaper can improve performance, by reducing the amount of data that is updated when users move items around the screen.

Command Line

Command option: `--disablewallpaper 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables the wallpaper.

```
--disablewallpaper 1
```

Displayed Soft Buttons

Usage: Select an option.

The following objects have this attribute:

- 3270 application
- 5250 application

Description

Specifies how many levels of “soft buttons” are displayed.

Command Line

Command option: `--3270b1 0|1|2|3|4`

Command option: `--b1 0|1|2|3|4`

Usage: Specify a level between 0 and 4.

The following example sets the number of levels of “soft buttons” for a 3270 application to 2.

```
--3270b1 2
```

The following example sets the number of levels of “soft buttons” for a 5250 application to 2.

```
--b1 2
```

Domain Name

Usage: Type the domain to use for application server authentication in the field.

Application server objects have this attribute.

Description

This attribute specifies the domain to use for the application server authentication process.

Note – This attribute plays no part in the SGD login.

Command Line

Command option: `--ntdomain dom`

Usage: Replace *dom* with the domain to use for application server authentication.

The following example authenticates using the domain indigo.

```
--ntdomain indigo
```


Email Address

Usage: Type the user's email address in the field.

User profile objects have this attribute.

Description

This attribute specifies a user's email address, in the form: *name@domain*

When authenticating users, SGD might use this attribute for identifying the user.

Command Line

Command option: `--email email`

Usage: Replace *email* with the user's email address.

The following example defines the email address of the user as `indigo@indigo-insurance.com`.

```
--email indigo@indigo-insurance.com
```

Emulation Type

Usage: Select an emulation type option.

Character application objects have this attribute.

Description

This attribute identifies the type of emulation required for the application: SCO Console, VT420, or Wyse 60. Set the correct [Terminal Type](#) for the selected Emulation Type.

Not all character application attributes apply to all emulation types. In the Administration Console, selecting an emulation type option enables and disables other attributes for the object.

Command Line

Command option: `--emulator scoconsole | vt420 | wyse60`

Usage: Specify the correct emulation type.

The following example uses Wyse 60 terminal emulation for the application.

```
--emulator wyse60
```

Environment Variables

Usage: Type the environment variables in the field, one on each line. Press Return to add new entries.

The following objects have this attribute:

- Character application
- X application
- 3270 application
- 5250 application

Description

This attribute specifies any environment variable settings needed to run the application. For example, you might need to set `LD_LIBRARY_PATH` to access shared libraries.

Quote any environment variable setting with a value containing spaces.

Do not set the `DISPLAY` variable. SGD sets the display automatically for each user.

Command Line

Command option: `--env setting`

Usage: Replace *setting* with an environment variable setting, of the form `VARIABLE=value`. To set more than one variable, use multiple `--env` arguments.

The following example runs the application with two environment variables set.

```
--env LD_LIBRARY_PATH=/usr/lib "MY_VARIABLE=603 1769"
```

Escape Sequences

Usage: Select an option.

Character application objects have this attribute.

Description

This attribute specifies how escape sequences are sent from the emulator to the application server. Escape sequences can be sent as 7-bit or 8-bit control codes.

This attribute applies to VT420 character applications only.

Command Line

Command option: `--escape 7-bit | 8-bit`

Usage: Specify a valid setting.

The following example sends escape sequences using 8-bit control codes.

```
--escape 8-bit
```

Euro Character

Usage: Select a setting from the list.

The following objects have this attribute:

- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute specifies the keycode mapping required by the application to support the euro character. Most euro-compliant applications currently use iso8859-15. If in doubt, check your X application's documentation to see which method to use.

To use the euro character with SGD, the client device must be capable of entering the character.

To display the euro character, you must configure your application to use an iso8859-15 font. Add one of the following to the [Arguments for Command](#) attribute:

```
-fn 5x7euro  
-fn 6x10euro  
-fn 6x13euro  
-fn 6x13boldeuro  
-fn 7x13euro  
-fn 7x13boldeuro  
-fn 7x14euro  
-fn 7x14boldeuro  
-fn 8x13euro  
-fn 8x13boldeuro  
-fn 8x16euro  
-fn 9x15euro  
-fn 9x15boldeuro  
-fn 10x20euro  
-fn 12x24euro
```

This ensures that the application uses the iso8859-15 fonts supplied with SGD. You can use your own fonts if you wish. However, to display the euro character they must be iso8859-15 compliant.

The application server must also support the euro character.

Command Line

Command option: `--euro unicode|iso8859-15`

Usage: Specify a valid option.

The following example enables iso8859-15 keycode mapping.

```
--euro iso8859-15
```

‘File’ and ‘Settings’ Menus

Usage: Select or deselect the check box.

The following objects have this attribute:

- 3270 application
- 5250 application

Description

Specifies whether or not the File and Settings menu items are enabled. When disabled, only the window resize buttons are displayed in the menu bar.

Command Line

Command option: `--3270si true|false`

Command option: `--si true|false`

Usage: Specify `true` or `false`.

The following example enables the File and Settings menu items for a 3270 application.

```
--3270si true
```

The following example enables the File and Settings menu items for a 5250 application.

```
--si true
```

Font Family

Usage: Select a font family from the list.

Character application objects have this attribute.

Description

This attribute determines the font family used within the terminal window for the application.

Only Courier, Helvetica, or Times Roman can be used. It is not possible to use any other font family.

Command Line

Command option: `--font courier | helvetica | timesroman`

Usage: Specify a valid font family.

The following example uses the Times Roman font in the application's terminal window.

```
--font timesroman
```

Font Size

Usage: Type a font size, in points, in the field.

Character application objects have this attribute.

Description

This attribute defines the font size in the terminal window, in the range 2-20 points.

Command Line

Command option: `--fontsize points`

Usage: Replace *points* with the font size in points.

The following example uses a 16-point font in the terminal window.

```
--fontsize 16
```

Font Size: Fixed Font Size

Usage: Select or deselect the Fixed Font Size check box.

Character application objects have this attribute.

Description

If this attribute is not selected, the emulator chooses a font size that fits the defined number of [Window Size: Columns](#) and [Window Size: Lines](#) into the [Window Size: Width](#) and [Window Size: Height](#) defined for the application. The application's [Font Size](#) setting is used as a minimum value.

If this attribute is selected, the [Font Size](#) defined is used, and scroll bars appear if necessary.

Note – If this attribute is selected, the [Window Size: Client's Maximum Size](#) attribute is ignored.

Command Line

Command option: `--fixedfont true|false`

Usage: Specify true or false.

The following example uses the font size specified by [Font Size](#) for the terminal window.

```
--fixedfont true
```

Font Smoothing

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

This attribute specifies whether to enable font smoothing for text in the Windows application. Enabling font smoothing improves text readability, but can affect performance.

Command Line

Command option: `--enablefontsmoothing 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example enables font smoothing for the Windows application.

```
--enablefontsmoothing 1
```

Foreground Color

Usage: Type a valid color resource, such as yellow, in the field.

The following objects have this attribute:

- 3270 application
- 5250 application

Description

Specifies the color of the text in the application's text window.

Color names are resolved to RGB values using the file named in the X Protocol Engine's RGB Database attribute.

Command Line

Command option: `--3270fg color`

Command option: `--fg color`

Usage: Replace *color* with a valid color resource, such as yellow.

In the following example, the text in the 3270 application's text window is set to the color plum4.

```
--3270fg plum4
```

In the following example, the text in the 5250 application's text window is set to the color plum4.

```
--fg plum4
```

Full Window Drag

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

This attribute specifies whether to show the contents of a window while it is moved in a Windows application. Disabling this feature can improve performance.

Command Line

Command option: `--disablefullwindowdrag 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables the option to show the contents of a window while it is moved.

```
--disablefullwindowdrag 1
```

Graphics Acceleration

Usage: Select or deselect the check box.

The following objects have this attribute:

- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute specifies whether acceleration is enabled. Acceleration optimizes how graphics are rendered and improves performance at the expense of smoothness and exactness. For example, colors might not always be exact.

If your application's display must always be exact, deselect the check box.

Command Line

Command option: `--accel true | false`

Usage: Specify true or false.

The following example enables graphics acceleration for the application's display.

```
--accel true
```

Hints

Usage: Type the hints in the field. Separate each hint with a semi-colon.

The following objects have this attribute:

- Character application
- Document
- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute enables you to define one or more strings that can be used to control the publishing and display of objects on the webtop.

You can use any number of strings and the strings can be anything. Separate each hint with a semi-colon. Use a name=value naming convention for webtop hints.

This attribute is blank by default.

This attribute is for developers who are using the SGD web services to develop custom webtops.

Command Line

Command option: `--hints hint...`

Usage: Replace *hint* with the webtop hint. Separate each hint with a semi-colon.

The following example sets a hint that might be used to specify the size of the webtop icon for the application.

```
--hints "preferredsize=16;"
```

Hosted Applications Tab

Usage: To assign applications to an application server object, click the Add button in the Editable Assignments table.

To delete applications for an application server object, use the Delete button in the Editable Assignments table.

The following objects have this attribute:

- Application server
- Dynamic application server

Description

The Hosted Applications tab lists the applications that are hosted by the application server.

The following sections of the Hosted Applications tab are used to display, select and assign applications:

- Effective Applications table
- Editable Assignments table

Effective Applications Table

The Effective Applications table shows all the application objects that are assigned to the selected object. The Local Assignments section of the table lists applications that are selected from the local repository.

The Assignment Type column shows one of the following:

- **Direct.** The assignment was made using the Editable Assignments table.
- **Indirect.** The assignment is the result of another relationship, such as membership of a group, or inheritance from another object.
- **Multiple.** The assignment has multiple sources, both Direct and Indirect.

If an assignment type is Indirect or Multiple, clicking the See Details link displays information that enables you to trace the origin of the link.

Editable Assignments Table

You can use the Editable Assignments table to select applications from the local repository.

Click the Add button in the Editable Assignments table. The Add Application Assignment window is shown.

To select applications in the Add Application Assignment window, do either of the following:

- **Browse the Navigation Tree.** As you browse the tree, the Content Area is updated with applications.
- **Use the Search Applications field.** Use this field to search for applications. Type in the names of applications in the field. Note that you can use the "*" wildcard in your search string. Typing a search string of *name* is equivalent to searching for "**name**" and returns any match of the search string. Results of the search are displayed in the Search Results table in the Content Area. The number of results returned by a search is limited to 150, by default.

Select the required applications from those listed in the Content Area. When you have finished selecting applications click the Add button.

The selected applications are displayed in the Effective Applications table of the Hosted Applications tab.

To delete applications from the Hosted Applications tab, use the Delete button in the Editable Assignments table.

Command Line

There is no command-line equivalent for this attribute.

Hosting Application Servers Tab

Usage: To assign application servers to a character, Windows, or X application object, click the Add button in the Editable Assignments table.

To delete application servers for a character, Windows, or X application object, use the Delete button in the Editable Assignments table.

The following objects have this attribute:

- Character application
- Windows application
- X application

Description

This attribute defines the application servers that can run the application. The SGD server uses application server load balancing to determine the application server to use. Each application server is stored as a *reference to the object*, so a particular object can appear on many Hosting Application Server tabs. If an object is moved or renamed later, all references to the object are automatically updated.

If a group is added to a Hosting Application Servers tab, the group's members and not the group are used for application server load balancing.

If you do not specify any application servers to run the application, the application can run on any SGD server in the array that supports that type of application.

The following sections of the Hosting Application Servers tab are used to display, select and assign applications:

- Effective Application Servers table
- Editable Assignments table

Effective Application Servers Table

The Effective Application Servers table shows all the application server objects that are assigned to the selected object. The Local Assignments section of the table lists applications that are selected from the local repository.

The Assignment Type column shows one of the following:

- **Direct.** The assignment was made using the Editable Assignments table.
- **Indirect.** The assignment is the result of another relationship, such as membership of a group, or inheritance from another object.
- **Multiple.** The assignment has multiple sources, both Direct and Indirect.

If an assignment type is Indirect or Multiple, clicking the See Details link displays information that enables you to trace the origin of the link.

Editable Assignments Table

You can use the Editable Assignments table to select application servers from the local repository.

Click the Add button in the Editable Assignments table. The Add Application Server Assignment window is shown.

To select application servers in the Add Application Server Assignment window, do either of the following:

- **Browse the Navigation Tree.** As you browse the tree, the Content Area is updated with application servers.
- **Use the Search Application Servers field.** Use this field to search for application servers. Type in the names of application servers in the field. Note that you can use the "*" wildcard in your search string. Typing a search string of *name* is equivalent to searching for **name** and returns any match of the search string. Results of the search are displayed in the Search Results table in the Content Area. The number of results returned by a search is limited to 150, by default.

Select the required application servers from those listed in the Content Area. When you have finished selecting application servers click the Add button.

The selected application servers are displayed in the Effective Application Servers table of the Hosting Application Servers tab.

To delete application servers from the Hosting Application Servers tab, use the Delete button in the Editable Assignments table.

Command Line

Command option: `--appserv object`

Usage: Replace *object* with the full name of an object, for example, "o=appservers/ou=IT/cn=london". Make sure that you quote any object names containing spaces.

The following example adds geneva and prague as application servers for an application.

```
--appserv "o=appservers/ou=IT/cn=geneva" \  
          "o=appservers/cn=prague"
```

Icon

Usage: Click the Edit button and select an icon option from the Select Application Icon list. Click OK to save the setting.

The following objects have this attribute:

- Character application
- Windows application
- X application
- Document
- 3270 application
- 5250 application
- Dynamic application

Description

This attribute specifies the icon that users see on their webtop, or their desktop Start menu or Launch menu.

Command Line

Command option: `--icon icon_name`

Usage: Replace *icon_name* with a file name, including the extension. For example, `spreadsheet.gif`.

The following example uses the `clock.gif` icon.

```
--icon clock.gif
```

Inherit Assigned Applications from Parent

Usage: Select or deselect the check box and click the Save button.

The following objects have this attribute:

- Organizational Unit
- User profile

Description

This attribute determines whether the assigned applications for the object also includes the assigned applications for the object's parent in the organizational hierarchy.

Depending on this attribute's setting in the parent object, the aggregation of assigned applications can continue up the hierarchy to the organization object.

Command Line

Command option: `--inherit true | false`

Usage: Specify `true` or `false`.

In the following example, the object inherits assigned applications from the parent object.

```
--inherit true
```

Interlaced Images

Usage: Select an option.

The following objects have this attribute:

- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute determines whether images are transmitted and displayed in a series of interlaced passes or in one pass from top to bottom.

Select Adjust Dynamically to allow interlacing to be turned on or off at any stage, according to the network conditions.

Use interlacing for graphics-intensive applications, particularly over low-bandwidth connections.

Command Line

Command option: `--interlaced automatic|on|off`

Usage: Specify a valid setting.

The following example enables interlaced image transmission.

```
--interlaced on
```

Keep Launch Connection Open

Usage: Select or deselect the check box.

The following objects have this attribute:

- X application
- 3270 application
- 5250 application

Description

This attribute specifies whether to keep open the connection used to start the application, or to close the connection.

Usually, you deselect the check box.

Select the check box if users experience either of these symptoms:

- The application appears to start and then immediately exits
- The application has problems shutting down. In this case, also set the [Session Termination](#) attribute to Login Script Exit

Command Line

Command option: `--keepopen true | false`

Usage: Specify `true` or `false`.

The following example closes the connection used to start the application.

```
--keepopen false
```

Keyboard Codes Modification

Usage: Select or deselect the check box.

Character application objects have this attribute.

Description

This attribute determines whether the application can change the codes generated by keys on the keyboard.

This attribute applies to Wyse 60 character applications only.

Command Line

Command option: `--appkeymode true|false`

Usage: Specify `true` or `false`.

The following example disables key code changes for the application.

```
--appkeymode false
```

Keyboard Map

Usage: For *user profile* objects, select an option. For the Custom Value option, type the path name of a keyboard map file in the field. For *character applications*, type the path name of a keyboard map file in the field.

The following objects have this attribute:

- User profile
- Character application

Description

This attribute specifies the path name of a keyboard map file. You can use a full path name or a relative path name. Relative path names are relative to the `/opt/tarantella/etc/data/keymaps` directory.

User Profile Objects

The keyboard map file specified is used for all graphical applications started by this user.

To use a keyboard map based on the locale of the client device, select Client's Input Locale. The actual keymap used is determined using the `/opt/tarantella/etc/data/keymaps/xlocales.txt` file.

Note – You can use the `*` or `?` wildcards in the `xlocales.txt` file to support a range of input locales. See the `xlocales.txt` file for details.

To use the X Protocol Engine settings defined for an SGD server to determine the keyboard map, select the X Protocol Engine Value option.

Alternatively, to always use a particular keyboard map for this user, type a file name.

Character Application Objects

The specified keyboard map file is used for this application.

Leave blank to use the default keyboard map for the application type. These are built-in to the emulators, but are equivalent to the keyboard maps in the files `ansikey.txt`, `vt420key.txt` and `w60key.txt`. These files are in the `/opt/tarantella/etc/data/keymaps` directory.

Command Line

Command option: `--keymap keymap`

Usage: For *user profile* objects, use either `default` or `client-locale` or replace *keymap* with the path name of a keyboard map file. For *character applications*, replace *keymap* with the path name of a keyboard map file.

The following example uses the named keymap, which is stored in `/opt/tarantella/etc/data/keymaps`.

```
--keymap mykeymap.txt
```

Keyboard Map: Locked

Usage: Select or deselect the check box.

The following objects have this attribute:

- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute specifies whether an application is prevented from changing the default keyboard mappings. Select the check box to ensure that the keyboard mappings cannot be changed.

Command Line

Command option: `--lockkeymap true | false`

Usage: Specify `true` or `false`.

The following example prevents an application from changing keyboard mappings.

```
--lockkeymap true
```

Keyboard Type

Usage: Select a keyboard type option.

The following objects have this attribute:

- 3270 application
- 5250 application

Description

Specifies the layout to use for mapping the keyboard to the terminal being emulated.

Command Line

Command option: `--3270kt pc|sun4|sun5|hp`

Command option: `--kt pc|sun4|sun5|hp`

Usage: Specify one of the valid keyboard types.

In the following example, the keyboard type for a 3270 application is set to pc.

```
--3270kt pc
```

In the following example, the keyboard type for a 5250 application is set to pc.

```
--kt pc
```

Kiosk Mode Escape

Usage: Select or deselect the check box.

The following objects have this attribute:

- Windows application
- X application

Description

Enables a pull-down header for the application. The header includes icons for minimizing and closing the application window. This attribute is only effective for applications having a [Window Type](#) setting of Kiosk mode.

To display the pull-down header when this attribute is enabled, move the mouse to the top of the application window.

Command Line

Command option: `--allowkioskescape true | false`

Usage: Specify true or false. The default setting is true.

The following example disables the pull-down header.

```
--allowkioskescape false
```

Line Wrapping

Usage: Select or deselect the check box.

Character application objects have this attribute.

Description

This attribute determines the behavior when a user types characters extending beyond the right edge of the terminal window.

Select the check box to wrap the characters onto the next line.

Deselect the check box to not display the characters. The characters are placed in the keyboard buffer.

Command Line

Command option: `--autowrap true|false`

Usage: Specify true or false.

The following example wraps characters onto the next line in the terminal window.

```
--autowrap true
```

Load Balancing Groups

Usage: Type one or more load balancing groups for the application server in the field. Press the Return key after each load balancing group.

Application server objects have this attribute.

Description

This attribute specifies the load balancing group used for application load balancing.

You can use any string, for example “Scandinavia” or “US-East”. Application load balancing tries to choose an application server and SGD server with the same location, to minimize the “network distance” between them and maximize performance. The connection between the user’s client device and the SGD server uses the AIP protocol, which adapts to the network conditions.

Leave this attribute blank unless you use an array spanning a wide area network (WAN), or one that includes slow links, and you use the intelligent array routing load balancing groups feature. More than one string is allowed, but this slows application startup.

If used, set this attribute on all appropriate application server objects, and for all SGD servers in the array. Use the Server Settings → General tab of the Administration Console.

Command Line

Command option: `--location location`

Usage: Replace *location* with the location of the application server.

The following example locates the application server in Paris.

```
--location Paris
```

Local Client Launch

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

This attribute specifies whether to try starting the application on the user's client device.

If this attribute is selected and the application is not installed on the client device, SGD then tries to start the application on an application server.

If this attribute is selected, the application is not resumable, even if an application server is used.

Command Line

Command option: `--trylocal true | false`

Usage: Specify true or false.

The following example tries to start the application on the client device.

```
--trylocal true
```

Login

Usage: Select or deselect the check box.

User profile objects have this attribute.

Description

This attribute specifies whether someone can log in using this user profile object.

Deselect the check box to deny a user access to SGD.

This attribute is always selected for profile objects in the System Objects organization. Users can always log in using the profile object, as long as the appropriate authentication mechanism is available. The authentication mechanism is configured on the Global Settings → Secure Global Desktop Authentication tab of the Administration Console.

To deny access to all users who use a particular authentication mechanism, deselect the appropriate authentication repository using the Authentication Wizard on the Global Settings → Secure Global Desktop Authentication tab of the Administration Console.

To stop all users from logging in to a particular SGD server, deselect [User Login](#) for the server on the Server Settings → General tab of the Administration Console.

Command Line

Command option: `--enabled true|false`

Usage: Specify true or false.

The following example enables the user profile object to log in to SGD.

```
--enabled true
```

Login: Multiple

Usage: Select or deselect the check box.

User profile objects have this attribute.

Description

This attribute specifies whether the user profile is used by a single user, or can be shared by multiple users in the form of a “guest” account.

The following table shows the similarities and differences between user profile objects with the attribute deselected and with the attribute selected.

| Account is Not Shared | Account is Shared |
|--|--|
| Must be used by one user. | Can be used by more than one user. |
| Each user has their own application sessions. | Each user has their own application sessions. |
| Application sessions can continue between user sessions. | Application sessions end when a user logs out. |
| One set of password cache entries. | One set of password cache entries, which is shared between all users. |
| The user can save entries in the password cache. | Users cannot save entries in the password cache. |
| If the user is already logged in, logging in again from a different client device relocates the user session. The old user session ends. | Logging in again creates a new user session. No existing user sessions are affected. |

Command Line

Command option: `--shared true | false`

Usage: Specify `true` or `false`.

The following example enables the user profile object to be shared by multiple users in the form of a “guest” account.

```
--shared true
```

Login Name

Usage: Type the user’s login name in the field.

User profile objects have this attribute.

Description

This attribute specifies the login name of a user. This is typically their UNIX system user name.

An authentication repository might use this attribute for identifying and authenticating users.

Command Line

Command option: `--user username`

Usage: Replace *username* with the user's login name.

The following example defines the login name as `indigo`.

```
--user indigo
```

Login Script

Usage: Type the login script file name in the field.

The following objects have this attribute:

- Character application
- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute specifies the login script that runs to start this application. Only change this attribute if you are having problems starting an application.

To configure SGD to choose a login script automatically, leave the setting blank.

You can use a full path name or a relative path name. Relative path names are considered relative to the value of the Execution Protocol Engine's [Login Script Directory](#) attribute.

The current working directory of the login script is the directory containing the script. If the script sources another script using a relative path name, the path name is considered relative to this directory.

Command Line

Command option: `--login script`

Usage: Replace *script* with the file name of the login script to use.

The following example uses the custom login script `my_login.exp` to start the application.

```
--login my_login.exp
```

Make Universal PDF Printer the Default

Usage: Select or deselect the check box.

The following objects have this attribute:

- Organization
- Organizational Unit
- User profile
- Windows application

Description

Sets the SGD Universal PDF printer as the client's default printer when printing from Windows applications.

This attribute is only available if the [Universal PDF Printer](#) is enabled.

This attribute can only be edited using the Administration Console if [Client Printing: Override](#) is enabled for the object.

By default, the Universal PDF printer is not the default printer. The setting is `false` on the command line.

The setting for this attribute overrides the following:

- The setting for a parent object in the organizational hierarchy.
- The default setting configured on Global Settings → Printing tab of the Administration Console, if no parent object configuration exists.
- For Windows application objects, this setting overrides the printing configuration for organization, organizational unit, or user profile objects. The order of precedence for printing configuration is: Windows application → user profile → organizational unit → organization.

For organization, organizational unit, and user profile objects, changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--pdfisdefault 1|0`

Usage: Specify 1 (true) or 0 (false).

The following example makes the Universal PDF printer the default printer when printing from a Windows application.

```
--pdfisdefault true
```

Make Universal PDF Viewer the Default

Usage: Select or deselect the check box.

The following objects have this attribute:

- Organization
- Organizational Unit
- User profile
- Windows application

Description

Sets the SGD Universal PDF Viewer printer as the client's default printer when printing from Windows applications.

By default, the Universal PDF Viewer printer is not the default printer. The setting is `false` on the command line.

This attribute is only available if the [Universal PDF Viewer](#) is enabled.

This attribute can only be edited using the Administration Console if [Client Printing: Override](#) is enabled for the object.

The setting for this attribute overrides the following:

- The setting for a parent object in the organizational hierarchy
- The default setting configured on the Global Settings → Printing tab of the Administration Console, if no parent object configuration exists

- For Windows application objects, this setting overrides the printing configuration for organization, organizational unit, or user profile objects. The order of precedence for printing configuration is: Windows application → user profile → organizational unit → organization.

For organization, organizational unit, and user profile objects, changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--pdfviewerisdefault 1|0`

Usage: Specify 1 (true) or 0 (false).

The following example makes the Universal PDF Viewer printer the default printer when printing from Windows applications.

```
--pdfviewerisdefault true
```

Mappings Tab

Usage: To create a new mapping for a dynamic application, click the Add button in the Editable Mappings table.

To delete mappings for a dynamic application, click the Delete button in the Editable Mappings table.

Dynamic application objects have this attribute.

Description

The Mappings tab lists the mappings for a dynamic application.

Use this tab to create and manage mappings between a type string and the name of an application object. The type string is displayed to users when they select which application to run.

Command Line

Command option: `--mapping mappings`

Usage: Replace *mappings* with one or more type-application mappings.

The following example specifies a mapping between type `linux` and the `gnome_desktop` application.

```
--mapping linux="o=applications/cn=gnome_desktop"
```

Maximum Count

Usage: Type a number in the field.

Application server objects have this attribute.

Description

This attribute specifies the maximum number of application sessions that can be run concurrently on the application server using SGD.

When the maximum number of application sessions is reached, SGD does not allow users to start any further applications hosted on the application server.

Command Line

Command option: `--maxcount count`

Usage: Replace *count* with the maximum number of application sessions.

The following example sets the maximum number of application sessions for the application server to 64.

```
--maxcount 64
```

Members Tab

Usage: To add group members to a group object, click the Add button in the Editable Assignments table.

To delete group members from a group object, use the Delete button in the Editable Assignments table.

Group objects have this attribute.

Description

The Members tab shows the members of the selected group object. You can only create groups of applications or groups of application servers.

A group can have many members, including other groups. Each member is stored as *a reference to the object*, so a particular object can be a member of many groups. If an object is moved or renamed later, all references to the object are automatically updated.

The following sections of the Members tab are used to display, select, and assign group members:

- Effective Members table
- Editable Members table

Effective Members Table

The Effective Members table shows all the objects that are assigned to the selected group object.

The Assignment Type column shows one of the following:

- **Direct.** The assignment was made using the Editable Assignments table.
- **Indirect.** The assignment is the result of another relationship, such as membership of a group, or inheritance from another object.
- **Multiple.** The assignment has multiple sources, both Direct and Indirect.

If an assignment type is Indirect or Multiple, clicking the See Details link displays information that enables you to trace the origin of the link.

Editable Members Table

You can use the Editable Members table to select group members from the local repository.

Click the Add button in the Editable Assignments table. The Add Application Member window, or Add Application Server Member window, is shown, depending on whether you are editing a group of applications or a group of application servers.

To select group members in the Add Application Assignment or Add Application Server Member window, do either of the following:

- **Browse the Navigation Tree.** As you browse the tree, the Content Area is updated with applications.

- **Use the Search Applications or Search Application Servers field.** The name of this field varies, depending on whether you are editing a group of applications or a group of application servers. Use this field to search for group members. Type in the names of applications or application servers in the field. Note that you can use the "*" wildcard in your search string. Typing a search string of *name* is equivalent to searching for "**name**" and returns any match of the search string. Results of the search are displayed in the Search Results table in the Content Area. The number of results returned by a search is limited to 150, by default.

Select the required group members from those listed in the Content Area. When you have finished selecting members click the Add button.

The selected group members are displayed in the Effective Members table of the Members tab.

To delete members from the Members tab, use the Delete button in the Editable Members table.

Command Line

Command option: `--member object`

Usage: Replace *object* with the full name of the object. For example, "`o=Indigo Insurance/ou=Finance/cn=XClaim`". Make sure that you quote any object names containing spaces.

The following example names Indigo Jones and Emma Rald as members.

```
--member "o=Indigo Insurance/cn=Indigo Jones" \  
         "o=Indigo Insurance/ou=Marketing/cn=Emma Rald"
```

Menu Animations

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

This attribute specifies whether to show transition effects for menus and tooltips in the Windows application. Disabling these menu animation effects can improve performance.

Command Line

Command option: `--disablemenuanimations 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables menu animation effects.

```
--disablemenuanimations 1
```

Menu Bar

Usage: Select or deselect the check box.

The following objects have this attribute:

- 3270 application
- 5250 application

Description

Specifies whether the application's menu bar is displayed or not.

Command Line

Command option: `--3270mb true|false`

Command option: `--mb true|false`

Usage: Specify true or false.

In the following example, the 3270 application's menu bar is enabled.

```
--3270mb true
```

In the following example, the 5250 application's menu bar is enabled.

```
--mb true
```

Middle Mouse Timeout

Usage: Type a timeout, in milliseconds, in the field.

The following objects have this attribute:

- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute enables you to emulate the middle mouse button on a two-button mouse by clicking the left and right mouse buttons at the same time.

This attribute is the maximum time that can elapse between pressing the left and the right mouse buttons for the action to be treated as a middle mouse button operation.

Command Line

Command option: `--middlemouse ms`

Usage: Replace *ms* with a timeout in milliseconds.

In the following example, the left and right buttons must be pressed within 0.3 seconds for the operation to be considered as a middle mouse button operation.

```
--middlemouse 300
```

Monitor Resolution

Usage: Type a resolution, in dots per inch, in the field.

The following objects have this attribute:

- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute specifies the monitor resolution, in dots per inch, that SGD reports to X applications asking for this information. Some X applications need this value to determine what font size to use.

If you leave this attribute blank, the value specified in the X Protocol Engine's [Monitor Resolution](#) attribute is reported.

The default resolution might cause the X application to choose a font size larger than it normally uses. This can cause clipping problems, as the X application needs more screen space. If this happens, try reducing the resolution by typing a smaller value, for example, 75.

The X application might also use too large a font if the X Protocol Engine's [Font Path](#) attribute uses a different order than the console or X terminal.

Command Line

Command option: `--dpi dpi`

Usage: Replace *dpi* with a resolution in dots per inch.

The following example reports a resolution of 75 dpi to X applications that need this information.

```
--dpi 75
```

Mouse

Usage: Select or deselect the Only 3-Button Mouse Supported check box.

X application objects have this attribute.

Description

This attribute enables you to specify whether the X application only supports a 3-button mouse.

Select the check box if the application only supports a 3-button mouse. The check box is cleared by default.

Command Line

Command option: `--force3button true|false`

Usage: Specify true or false.

In the following example, the application only supports a 3-button mouse.

```
--force3button true
```

Name

Usage: Type the name used for the object, for example, Indigo Jones.

The following objects have this attribute:

- Active Directory container
- Character application
- Document
- Domain component
- Group
- Application server
- User profile
- Windows application
- X application
- 3270 application
- 5250 application
- Organization
- Organizational Unit
- Dynamic application
- Dynamic application server

Description

This attribute specifies the name of the object in the local repository.

The following naming conventions are used for SGD objects.

- 3270 application objects have a `cn=` naming attribute.
- 5250 application objects have a `cn=` naming attribute.

- Active Directory container objects have a `cn=` naming attribute.
- Application server objects have a `cn=` naming attribute.
- Dynamic application server objects have a `cn=` naming attribute.
- Character application objects have a `cn=` naming attribute.
- Document objects have a `cn=` naming attribute.
- Domain Component objects have a `dc=` naming attribute.
- Group objects have a `cn=` naming attribute.
- Organization objects have an `o=` naming attribute.
- Organizational Unit objects have an `ou=` naming attribute.
- User profile objects can have a `cn=` (common name), a `uid=` (user identification), or a `mail=` (mail address) naming attribute.
- Windows application objects have a `cn=` naming attribute.
- X application objects have an `cn=` naming attribute.
- Dynamic application objects have an `cn=` naming attribute.

In the Administration Console, names can include any characters, except the backslash (\) or plus (+) characters.

When you create a new application server object, the Name setting is automatically entered in the Address field.

Command Line

Command option: `--name name`

Usage: Replace *name* with the full name of the object, for example, `"o=applications/ou=Finance/cn=XClaim"`.

Make sure that you quote any names containing spaces.

If you use a forward slash (/) in an object name, you must backslash protect (escape) it. For example, to create an object with the relative name `cn=a/b` beneath `o=organisation`, type `cn=a\b`.

This creates an object `o=organisation/"cn=a/b"`.

The following example defines the name of the organization object as Indigo Insurance.

```
--name "o=Indigo Insurance"
```

The following example defines the name of the organizational unit object as Finance. The object belongs to the directory object, Indigo Insurance, which must already exist.

```
--name "o=Indigo Insurance/ou=Finance"
```

The following example defines the common name of a user profile object as Indigo Jones. The object belongs to the organization object, Indigo Insurance.

```
--name "o=Indigo Insurance/cn=Indigo Jones"
```

The following example defines the names of a domain component object as indigo-insurance.

```
--name "dc=com/dc=indigo-insurance"
```

Number of Sessions

Usage: Select or deselect the Limited check box. If the Limited check box is selected, type a number in the Max per User field.

The following objects have this attribute:

- Character application
- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute enables you to set the maximum number of instances of an application a user can run simultaneously. The default is 3.

The application's link on the webtop indicates how many instances of the application the user can run. The webtop also provides tools for suspending, resuming or ending each application instance.

Command Line

Command option: `--maxinstances 0 | instances`

Usage: Specify 0 or replace *instances* with the number of instances.

The following example sets the maximum number of instances of the application to unlimited.

```
--maxinstances 0
```

Numpad Codes Modification

Usage: Select a keypad behavior option from the list.

Character application objects have this attribute.

Description

This attribute specifies the behavior of the numeric keypad, whether it always generates numbers or whether you want the application to change the codes generated by the keypad.

This attribute applies to VT420 character applications only.

Command Line

Command option: `--keypad numeric | application`

Usage: Specify the keypad behavior you want.

In the following example, the keypad always generates numbers.

```
--keypad numeric
```

Passwords Tab

Usage: Use the Password Cache table to manage entries in the password cache.

The following objects have this attribute:

- Application server
- Dynamic application server
- User profile

Description

The Passwords tab lists the password cache entries for the selected user profile or application server object.

Use the New button to add a password cache entry, using the Create New Password Cache Entry page.

Use the Edit button to edit an entry in the password cache, or the Delete button to remove an entry from the password cache.

Use the Reload button to refresh the Password Cache table.

Use the Search field to search for entries in the Password Cache table. You can use the "*" wildcard in your search string. Typing a search string of *name* is equivalent to searching for "**name**" and returns any match of the search string. The number of results returned by a search is limited to 150, by default.

Command Line

On the command line, use the `tarantella passcache` commands to delete and examine entries in the password cache. See ["The tarantella passcache Command" on page 762](#).

Password Cache Usage

Usage: Select the Override Global Setting check box and then select or deselect the Secure Global Desktop Password Tried option. To use the default setting defined in the Global Settings → Application Authentication tab, deselect the Override Global Setting check box.

The following objects have this attribute:

- Application server
- Dynamic application server

Description

This attribute specifies the policy for authenticating users on the application server, *if no password is already cached* for that server.

The Administration Console options and their command line equivalents are shown in the following table.

| Administration Console | Command Line | Description |
|--|---------------------------------|--|
| Secure Global Desktop Password Tried (selected) | <code>--auth trytta</code> | If the user's password for logging in to SGD is cached, the same password is used to try to log in to the application server. If the attempt fails, the user is prompted for a password. When listing object attributes on the command line, this attribute value is displayed as <code>true</code> . |
| Secure Global Desktop Password Tried (deselected) | <code>--auth nevertrytta</code> | The user's password for logging in to SGD is not used. The user is prompted to enter a user name and password for the application server. When listing object attributes on the command line, this attribute value is displayed as <code>false</code> . |
| Override Global Setting (deselected) | <code>--auth default</code> | The Password Cache Usage attribute determines whether to try the user's password or not. When listing object attributes on the command line, this attribute value is displayed by <code>default</code> . |

A user's password for logging in to SGD can be stored in the password cache if an SGD server is also used as an application server, or if [Password Cache](#) is selected in the [Secure Global Desktop Authentication Tab](#).

Command Line

Command option: `--auth trytta|nevertrytta|default`

Usage: Specify one of the valid settings.

The following example tries the password the user typed to log in to SGD, if the password is cached.

```
--auth trytta
```

Postscript Printer Driver

Usage: Type the name of the printer driver to use for PDF printing in the field.

The following objects have this attribute:

- Organization
- Organizational Unit

- User profile
- Windows application

Description

The name of the printer driver to use for PDF printing when printing from Windows applications.

This printer driver must be installed on every Windows application server used with SGD.

The printer driver must be a PostScript™ printer driver. The default is HP Color LaserJet 2800 Series PS.

The name you type must match the name of the printer driver installed on your Windows application servers *exactly*. Pay particular attention to the use of capitals and spaces. The `/opt/tarantella/etc/data/default.printerinfo.txt` file contains all the common printer driver names ordered by manufacturer. To avoid errors, copy and paste the driver name from this file.

This attribute is only available if [Universal PDF Printer](#) is enabled.

This attribute can only be edited using the Administration Console if [Client Printing: Override](#) is enabled for the object.

The setting for this attribute overrides the following:

- The setting for a parent object in the organizational hierarchy.
- The default setting configured on the Global Settings → Printing tab of the Administration Console, if no parent object configuration exists.
- For Windows application objects, this setting overrides the printing configuration for organization, organizational unit, or user profile objects. The order of precedence for printing configuration is: Windows application → user profile → organizational unit → organization.

For organization, organizational unit, and user profile objects, changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--pdfdriver driver_name`

Usage: Replace *driver_name* with the name of the printer driver to use for PDF printing. Use quotes on the command line if the name includes spaces.

The following example configures the HP LaserJet 8000 Series PS printer driver as the driver to use for PDF printing.

```
--pdfdriver "HP LaserJet 8000 Series PS"
```

Printer Preference Caching

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

This attribute disables the caching of printer preferences, such as page orientation and paper size, for a Windows application.

Command Line

Command option: `--noprintprefs 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables the caching of printer preferences.

```
--noprintprefs 1
```

Prompt Locale

Usage: Type a locale in the field.

Application server objects have this attribute.

Description

This attribute controls the language used in the login scripts when pattern matching the login data from an application server.

When using the login scripts supplied with SGD, the `vars.exp` script defines variables for matching system prompts. By default, English system prompts are supported. This script can be customized to support users in other locales.

A locale has two parts, a *language* and an optional *territory*, separated by an underscore.

The language part of a locale is specified using ISO 639 language codes, for example `en` for English or `ja` for Japanese.

The territory part of a locale is specified using ISO 3166 territory codes, for example `us` for the United States or `jp` for Japan.

By default, the locale is `en_us`.

Command Line

Command option: `--hostlocale ll_tt`

Usage: Replace `ll_tt` with a locale.

The following example sets the default language of the application server object to French. French prompts must be configured in the login scripts used with this application server.

```
--locale fr
```

Remote Audio

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

This attribute configures whether audio is sent from the Windows application server to the client device. If Remote Audio is enabled, audio is played on the Windows application server.

Enabling this attribute has the same effect as the “Leave at remote computer” sound setting for a Microsoft Windows Remote Desktop connection.

By default, this attribute is disabled.

Command Line

Command option: `--remoteaudio 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example keeps remote audio at the Windows application server.

```
--remoteaudio 1
```

Scroll Style

Usage: Select a scroll style option.

Character application objects have this attribute.

Description

This attribute specifies how the terminal window scrolls. The available options are line-by-line, several lines at once, or smoothly.

When listing object attributes on the command line, the following applies:

- The `line` attribute value is displayed as `normal`
- The `multiple` attribute value is displayed as `jump`

Command Line

Command option: `--scrollstyle line | multiple | smooth`

Usage: Specify the scroll style you want.

The following example scrolls the terminal window smoothly.

```
--scrollstyle smooth
```

Serial Port Mapping

Usage: For *user profile* objects or *organizational unit* objects, select the Override Parent's Setting check box and then select or deselect the Enabled option. To use the setting defined for the parent object, deselect the Override Parent's Setting check box.

For *organization* objects, select the Override Global Setting check box and then select or deselect the Enabled option. To use the setting defined in the Global Settings tab, deselect the Override Global Setting check box.

The following objects have this attribute:

- Organization
- Organizational Unit
- User profile

Description

This attribute controls whether users can access the serial ports on a client device from a Windows Terminal Services session.

By default, a user profile object or organizational unit object inherits the setting of its parent object in the organizational hierarchy. This is used to enable or disable access to serial ports for many users without having to edit each user profile object. To override this, select the Override Parent's Setting check box and change the setting.

By default, organization objects use the global setting configured on the Global Settings → Client Device tab of the Administration Console. To override this, select the Override Global Setting check box and change the setting.

The Administration Console options and their command line equivalents are shown in the following table.

| Administration Console | Command Line | Description |
|--|--------------|---|
| Override Parent's Setting (deselected) | 2 | User profile objects or organizational unit objects. Use the setting inherited from the parent object. This is the default setting. |
| Override Global Setting (deselected) | 2 | Organization objects. Use the global setting. This is the default setting. |
| Enabled (selected) | 1 | Enable access to serial ports. |
| Enabled (deselected) | 0 | Disable access to serial ports. |

When a user starts a Windows application, SGD checks the user profile object for the user and then any parent object further up the organizational hierarchy to see whether access to serial ports is enabled or disabled. If all the objects selected are configured to use the parent's setting, then the default setting is used.

By default, access to serial ports is enabled.

Command Line

Command option: `--serialport 2|1|0`

Usage: Specify 2|1|0.

The following example disables access to serial ports.

```
--serialport 0
```

Server Address

Usage: Type DNS name or IP address of the application server in the field.

The following objects have this attribute:

- 3270 application
- 5250 application

Description

This attribute names the 3270 (mainframe) or AS/400 application server that runs the application.

Use a DNS name rather than an IP address, if you know the DNS name.

Command Line

Command option: `--hostname host`

Usage: Replace *host* with the DNS name or IP address of the 3270 (mainframe) or AS/400 application server.

The following example runs the application on the application server warsaw.indigo-insurance.com.

```
--hostname warsaw.indigo-insurance.com
```

Server Port

Usage: Type the Transmission Control Protocol (TCP) port number used to connect to the application server in the field.

The following objects have this attribute:

- 3270 application
- 5250 application

Description

This attribute specifies the TCP port used by the emulator to exchange data with the 3270 (mainframe) application server or AS/400 application server.

By default, TCP port 23 is used.

Command Line

Command option: `--portnumber tcp`

Usage: Replace *tcp* with the TCP port number used to connect to the application server.

The following example connects on TCP port 4567 to the application server.

```
--portnumber 4567
```

Session Termination

Usage: Select a setting from the list.

The following objects have this attribute:

- X application
- 3270 application
- 5250 application

Description

This attribute determines when an application session ends.

The Administration Console options and their command line equivalents are shown in the following table.

| Administration Console | Command Line | Description |
|---|-----------------------------------|---|
| Last Client Exit | <code>lastclient</code> | The SGD server keeps track of the number of X clients running within the session, and ends the session when this reaches zero. |
| Window Manager Exit | <code>windowmanager</code> | The SGD server ends the session when the Window Manager exits, no matter how many X clients are running. |
| Only Window Manager Remaining | <code>windowmanageralone</code> | The SGD server ends the session when the only remaining X client is the Window Manager. Some Window Managers, such as OpenLook, run X clients in the background, which means that this condition is never met. If you encounter this problem, use the No Visible Windows setting. |
| Login Script Exit | <code>loginscript</code> | The SGD server ends the session when the login script completes. Use this setting with Keep Launch Connection Open if an application has problems shutting down. |
| No Visible Windows | <code>nowindows</code> | The SGD server ends the session when no windows are visible. This is useful for window managers, such as OpenLook, that run X clients in the background. |
| Login Script Exit or No Visible Windows | <code>loginscriptnowindows</code> | The SGD server ends the session when either the login script completes or no windows are visible. Use this setting for applications that have a General Application Resumability setting and that use X clients, as this forces a session to close if an application server is rebooted or disconnected from the network. Use this setting with Keep Launch Connection Open if an application has problems shutting down. |

Command Line

Command option: `--endswhen lastclient | windowmanager | windowmanageralone | loginscript | nowindows | loginscriptnowindows`

Usage: Specify a valid setting.

The following example ends the application session when no windows are visible.

```
--endswhen nowindows
```

SGD Remote Desktop Client

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

This attribute specifies whether to use the SGD Remote Desktop Client (`ttatssc`) to run an application from a Microsoft Windows application server, using the Microsoft RDP protocol.

If you deselect the check box, the [Local Client Launch](#) check box is automatically selected. Only deselect the check box, if you only want to run the Windows application on the client device.

Use the [Arguments](#) attribute to specify command-line options for the Remote Desktop Client.

Command Line

Command option: `--winproto wts | none`

Usage: Specify a valid setting.

The following example connects to a Microsoft Windows server using the Microsoft RDP protocol.

```
--winproto wts
```

Share Resources Between Similar Sessions

Usage: Select or deselect the check box.

The following objects have this attribute:

- X application
- 3270 application
- 5250 application

Description

This attribute specifies whether application sessions for applications configured with a [Window Type](#) setting of Client Window Management try to share resources. Sharing sessions reduces the memory overhead on both the SGD server and the client device.

Resources are shared between applications with the same settings for the following attributes:

- [Window Color: Custom Color](#)
- [Window Color](#)
- [Interlaced Images](#)
- [Graphics Acceleration](#)
- [Delayed Updates](#)
- [Middle Mouse Timeout](#)
- [Monitor Resolution](#)

Command Line

Command option: `--share true | false`

Usage: Specify true or false.

The following example enables resource sharing for similar sessions.

```
--share true
```

Status Line

Usage: Select a type of status line from the list.

Character application objects have this attribute.

Description

This attribute specifies the type of status line to show for the application.

| Application Type | Types of Status Line Available |
|------------------|--|
| VT420 | <ul style="list-style-type: none">• None• Cursor Position and Print Mode• Messages from the Host |
| Wyse 60 | <ul style="list-style-type: none">• None• Standard• Extended |
| SCO Console | <ul style="list-style-type: none">• <i>Not Applicable</i> |

When listing object attributes on the command line, the attribute value `hostmessages` is displayed as `host writable`.

Command Line

Command option: `--statusline none | indicator | hostmessages | standard | extended`

Usage: Specify the type of status line you want. Not all settings are valid for all types of character application.

The following example does not display a status line.

```
--statusline none
```

Surname

Usage: Type the user's surname in the field.

User profile objects have this attribute.

Description

This attribute specifies the surname, or family name, of the user.

Names can include any characters you want.

Command Line

Command option: `--surname name`

Usage: Replace *name* with the surname of the user. Make sure that you quote any names containing spaces.

The following example defines the surname of the user as Jones.

```
--surname Jones
```

SWM Local Window Hierarchy

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

This attribute is only effective for applications having a [Window Type](#) setting of Seamless Window.

The attribute is needed for compatibility with some Borland applications. Enable the attribute if you are having trouble with minimizing and maximizing the application window from the task bar.

Command Line

Command option: `--swmopts 1|0`

Usage: Specify 1 (true) or 0 (false). The default setting is 0.

The following example enables SWM local window hierarchy for the application.

```
--swmopts 1
```

Terminal Type

Usage: Select a terminal type option, or select the Custom option and type in the field.

Character application objects have this attribute.

Description

This attribute specifies the terminal type required for the application. You must set this appropriately for the [Emulation Type](#).

Command Line

Command option: `--termtype type`

Usage: Replace *type* with a terminal type, for example, `ansi`.

The following example uses the `ansi` terminal type.

```
--termtype ansi
```

The following example uses the `wyse60` terminal type.

```
--termtype wyse60
```

Theming

Usage: Select or deselect the check box.

Windows application objects have this attribute.

Description

This attribute specifies whether to enable themes for a Windows application, such as a Windows desktop session. Disabling themes can improve performance.

Command Line

Command option: `--disabletheming 1 | 0`

Usage: Specify 1 (true) or 0 (false).

The following example disables themes.

```
--disabletheming 1
```

Tokens Tab

Usage: Use the Token Cache table to manage entries in the token cache.

User profile objects have this attribute.

Description

The Tokens tab is used to manage tokens used for the authentication token authentication mechanism. This authentication mechanism is used when the SGD Client is in Integrated mode.

The Tokens tab shows the token cache entries for the selected user profile object.

Use the Delete button to delete a token from the token cache.

Use the Reload button to refresh the Token Cache table.

Use the Search field to search for entries in the Token Cache table. You can use the `"*"` wildcard in your search string. Typing a search string of `name` is equivalent to searching for `"*name*"` and returns any match of the search string. The number of results returned by a search is limited to 150, by default.

Command Line

On the command line, use the `tarantella tokencache` commands to delete and examine entries in the token cache. See [“The tarantella tokencache Command” on page 836](#).

Use the `tarantella tokencache list` command to display entries in the token cache.

Command option: `tarantella tokencache list`

The following example lists all entries in the token cache.

```
tarantella tokencache list
```

Universal PDF Printer

Usage: Select or deselect the check box.

The following objects have this attribute:

- Organization

- Organizational Unit
- User profile
- Windows application

Description

This attribute enables users to print using the SGD Universal PDF printer when printing from Windows applications.

This attribute can only be edited using the Administration Console if [Client Printing: Override](#) is enabled for the object.

The setting for this attribute overrides the following:

- The setting for a parent object in the organizational hierarchy.
- The default setting configured on the Global Settings → Printing tab of the Administration Console, if no parent object configuration exists.
- For Windows application objects, this setting overrides the printing configuration for organization, organizational unit, or user profile objects. The order of precedence for printing configuration is: Windows application → user profile → organizational unit → organization.

For organization, organizational unit, and user profile objects, changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--pdfenabled 1|0`

Usage: Specify 1 (true) or 0 (false).

The following example enables users to print using the Universal PDF printer.

```
--pdfenabled 1
```

Universal PDF Viewer

Usage: Select or deselect the check box.

The following objects have this attribute:

- Organization
- Organizational Unit

- User profile
- Windows application

Description

This attribute enables users to print using the SGD Universal PDF Viewer printer when printing from Windows applications.

This attribute can only be edited using the Administration Console if [Client Printing: Override](#) is enabled for the object.

The setting for this attribute overrides the following:

- The setting for a parent object in the organizational hierarchy.
- The default setting configured on the Global Settings → Printing tab of the Administration Console, if no parent object configuration exists.
- For Windows application objects, this setting overrides the printing configuration for organization, organizational unit, or user profile objects. The order of precedence for printing configuration is: Windows application → user profile → organizational unit → organization.

For organization, organizational unit, and user profile objects, changes to this attribute only take effect for new user sessions.

Command Line

Command option: `--pdfviewerenabled 1|0`

Usage: Specify 1 (true) or 0 (false).

The following example enables users to print using the Universal PDF Viewer printer.

```
--pdfviewerenabled true
```

URL

Usage: Type a URL in the field.

Document objects have this attribute.

Description

The URL associated with the object. This is displayed when users click the link on their webtop or in their desktop Start or Launch menu

You can use absolute or relative URLs. Relative URLs are considered relative to the SGD document root. This is usually `/opt/tarantella/var/docroot`.

Command Line

Command option: `--url url`

Usage: Replace *url* with a URL. Make sure that you quote any values containing spaces or other characters that might be interpreted by your shell.

The following example makes the object display the Indigo Insurance home page when clicked.

```
--url http://www.indigo-insurance.com
```

The following example displays the specified URL, considered relative to the SGD document root.

```
--url ../my_docs/index.html
```

User Assignment

Usage: Type one or more search specifications in the field.

Application server objects have this attribute.

Description

Use this attribute to specify the users that can run applications on the application server.

The search specification filters application servers based on the user identity (the fully-qualified user name) of the user. The search filter can be any of the following:

- An RFC2254-compliant LDAP search filter
- An RFC1959-compliant LDAP URL
- A `scottasessionowner= filter`

Note – LDAP-based search filters are applied even if the user identity is not an LDAP identity.

For an RFC2254 search filter, enclose each filter in double quotes and brackets.

For an LDAP URL, use the format `ldap:///search-criteria`. If you include the host, port and return attribute specification in the URL they are ignored. This is because the LDAP directory server configured as part of SGD authentication is used.

Note – The Administration Console does not automatically escape the special characters specified in [RFC2254](#). To use a special character in the Administration Console, you must manually type the escape sequence. For example, to search for a user with the common name “John Doe (123456)”, type the following `cn=John Doe\0x28123456\0x29` in the search field. The command line does escape the special characters.

SGD supports the use of extensible matching search filters as specified in [RFC2254](#). This enables you to look up information from components that make up an object’s distinguished name (DN). For example, to assign an application server to a user that is contained within any OU called managers (`ou=managers`), you can use a `(&(ou:dn:=managers))` search filter.

The `scottasessionowner=` filter is a simple search that takes only the asterisk (*) wildcard.

Command Line

Command option: `--userassign spec`

Usage: Replace *spec* with a search specification.

The following example assigns the application server to any manager in the Sales department *and* anyone who has Violet Carson as their manager.

```
--userassign "(&(job=manager)(dept=Sales))" \
"(manager=Violet Carson)"
```

The following example assigns the application server to any manager in the Sales department of example.com.

```
--userassign "ldap:///ou=Sales,dc=example,dc=com??sub?job=manager"
```

The following example assigns the application server to users with the UID “violet” in the Sales department and the UID “emmarald” in the Marketing department.

```
--userassign uid=violet,ou=Sales,dc=example,dc=com \
uid=emmarald,ou=Marketing,dc=example,dc=com
```

The following example assigns the application server to all users in the Sales OU in the local repository.

```
--userassign "scottasessionowner=dc=com/dc=example/ou=Sales/*"
```

The following example assigns the application server to Elizabeth Blue in the Sales OU in the local repository.

```
--userassign "scottasessionowner=dc=com/dc=example/ou=Sales/cn=Elizabeth Blue"
```

User Sessions Tab

Usage: Use the buttons in the User Sessions tab to view and manage user sessions.

User profile objects have this attribute.

Description

This tab lists the active user sessions for the selected user profile object. A user session represents a user who is connected to an SGD server.

Use the View Details button in the User Session List table to show more details for the selected user session. Use the End button to end the selected user session. The Reload button refreshes the User Session List table.

Use the Search options to search the User Session List table. When searching for a User Identity or Secure Global Desktop Server, you can use the “*” wildcard in your search string. Typing a search string of *name* is equivalent to searching for “**name*” and returns any match of the search string.

To search for a Login Time, use a search string format of *yyyy/mm/dd hh:mm:ss*.

The number of results returned by a search is limited to 150, by default.

Command Line

On the command line, use the `tarantella webtopsession` commands to list and end user sessions. See [“The tarantella webtopsession Command” on page 850](#).

Use the `tarantella webtopsession list` command to show user session details for a specified user profile object.

Command option: `tarantella webtopsession list --person pobj`

Usage: Replace *pobj* with the full name of the user profile object.

The following example lists user sessions for the Indigo Jones user profile object.

```
tarantella webtopsession list \  
"o=Indigo Insurance/ou=IT/cn=Indigo Jones"
```

Virtual Server Broker Class

Usage: Select a broker from the list, or select Other from the list and type a fully-qualified class name in the field.

Dynamic application server objects have this attribute.

Description

The virtual server broker (VSB) interface class to use for the dynamic application server.

The following table shows the VSBs supplied with SGD.

| Broker | Fully-Qualified Class Name |
|------------------|--|
| SGD | <code>com.tarantella.tta.webservices.vsbim.SGD</code> |
| User-defined SGD | <code>com.tarantella.tta.webservices.vsbim.UserDefinedSGD</code> |
| VDI | <code>com.sun.sgd.vsbim.SunVDIVirtualServerBroker</code> |

This attribute can be used by developers who are using SGD web services to develop applications.

Command Line

Command option: `--vsbclass class-name`

Usage: Replace *class-name* with the fully-qualified class name for the VSB.

The following example uses the VDI broker.

```
--vsbclass com.sun.sgd.vsbim.SunVDIVirtualServerBroker
```

Virtual Server Broker Parameters

Usage: Type parameters for the virtual server broker (VSB) in the field.

Dynamic application server objects have this attribute.

Description

This attribute specifies a parameter string to pass to the VSB.

This attribute is blank by default.

This attribute can be used by developers who are using SGD web services to develop applications.

Command Line

Command option: `--vsbparams string`

Usage: Replace *string* with a string of parameters to pass to the VSB.

Window Close Action

Usage: Select a setting from the list.

The following objects have this attribute:

- Character application
- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute determines what happens if the user closes the main application window using the Window Manager decoration. This attribute only applies for applications that are configured with a [Window Type](#) setting of Client Window Management or Independent Window.

The Administration Console options and their command line equivalents are shown in the following table.

| Administration Console | Command Line | Description |
|------------------------|------------------------|--|
| Notify Application | <code>notifyapp</code> | <p>The application is notified of a close action in the normal way. If the application ignores the request, SGD kills the application.</p> <p>When listing object attributes on the command line, this attribute value is displayed as <code>notifyclient</code>.</p> <p>This setting only applies to X applications that are configured with a Window Type setting of Client Window Management.</p> |

| Administration Console | Command Line | Description |
|-----------------------------|-----------------------------|---|
| Kill Application | <code>killapp</code> | <p>SGD kills the application. This is similar to using the program <code>xkill</code> to exit the application. Use this setting only if your users are having difficulty closing an application.</p> <p>When listing object attributes on the command line, this attribute value is displayed as <code>killclient</code>.</p> <p>This setting only applies to X applications that are configured with a Window Type setting of Client Window Management.</p> |
| Suspend Application Session | <code>suspendsession</code> | <p>If the application object is resumable, the application's application session is suspended. If the application object is not resumable, the application session ends. Use this setting only if the application provides its own mechanism for the user to exit. See also Application Resumability.</p> <p>If you are using the SGD Client in Integrated mode, there are no controls for resuming a suspended application. Users have to log out and log in again to resume their applications, or display a webtop.</p> <p>For applications that are configured with a Window Type setting of Client Window Management, a confirmation prompt is displayed when the user closes the application using the window decoration.</p> |
| End Application Session | <code>endsession</code> | <p>SGD ends the application session.</p> <p>This is the default setting for Windows and character applications configured with a Window Type setting of Independent Window.</p> <p>For applications that are configured with a Window Type setting of Client Window Management, a confirmation prompt is displayed when the user closes the application using the window decoration.</p> |

Note – An application session can contain several main application windows, for example, a CDE session with several applications running. If this attribute is set to either Suspend Application Session or End Application Session, then closing any of the applications results in the entire session being suspended or ended.

Command Line

Command option: `--windowclose notifyapp | killapp | suspendsession | endsession`

Usage: Specify a valid setting.

In the following example, closing the application’s main window suspends the application session, as long as the application object is resumable.

```
--windowclose suspendsession
```

Window Color

Usage: Select an option. For the Custom Color option, type a color in the field.

The following objects have this attribute:

- X application
- 3270 application
- 5250 application

Description

This attribute determines the appearance of the root window.

Select Default Colors to show the standard X “root weave” pattern. To use your own color, select Custom Color and specify a [Window Color: Custom Color](#) attribute.

When listing object attributes on the command line, the `custom` attribute value is displayed as `color`.

Command Line

Command option: `--roottype default|custom`

Usage: Specify a valid setting.

The following example uses a custom color, which is specified using `--rootcolor`, for the root window.

```
--roottype custom
```

Window Color: Custom Color

Usage: Used when the Custom Color option is selected for the [Window Color](#) attribute. Type a valid color resource, such as `yellow`, in the field.

The following objects have this attribute:

- X application
- 3270 application
- 5250 application

Description

This attribute determines the color of the root window.

Color names are resolved to RGB values using the file named in the X Protocol Engine's RGB Database attribute.

Command Line

Command option: `--rootcolor color`

Usage: Replace *color* with a valid color resource, such as yellow.

In the following example, the root window uses the color plum4.

```
--rootcolor plum4
```

Window Management Keys

Usage: Select or deselect the check box.

The following objects have this attribute:

- Windows application
- X application

Description

Keyboard shortcuts that deal with window management can either be sent to the remote session or acted on locally. This attribute is only effective for applications having a [Window Type](#) setting of Kiosk mode.

To exit kiosk mode when this attribute is enabled, use the key sequence Alt-Ctrl-Shift-Space. This minimizes the kiosk session on the local desktop.

Command Line

Command option: `--remotewindowkeys 1 | 0`

Usage: Specify 1 (true) or 0 (false). The default setting is 0.

The following example sends window management keys to the remote session.

```
--remotewindowkeys 1
```

Window Manager

Usage: Type the full path name of the Window Manager in the field. Press Return to add new entries.

The following objects have this attribute:

- X application
- 3270 application
- 5250 application

Description

This attribute specifies the Window Manager to use for the application. You can also use this to name any other applications to run alongside the main application.

You can name as many Window Manager applications as you want.

A Window Manager is not needed for X applications configured with a [Window Type](#) setting of Client Window Management.

Command Line

Command option: `--winmgr command`

Usage: Replace *command* with a full path name. Separate each path name with a space.

The following example runs the application using the twm Window Manager.

```
--winmgr /usr/local/bin/twm
```

Window Size: Client's Maximum Size

Usage: Select or deselect the check box.

The following objects have this attribute:

- Character application
- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute affects the initial size of the application.

Select the check box to ensure that the application fills the user's screen when it starts.

The application appears with window decoration. To cause an application to fill the screen completely, without window decoration, set the application object's [Window Type](#) attribute to Kiosk.

Deselect the check box to size the application according to the object's [Window Size: Width](#) and [Window Size: Height](#) attributes.

Unless [Window Size: Scale to Fit Window](#) is selected, the application size does not change during the lifetime of the application session. If the user starts an application on one client device, then resumes the same application on a client device with a different screen resolution, the application does not resize to fit the screen.

Note – If this attribute is selected and the application is a character application, the [Font Size: Fixed Font Size](#) attribute *must* be deselected.

Command Line

Command option: `--maximize true | false`

Usage: Specify `true` or `false`.

The following example displays the application at maximum size on the client device.

```
--maximize true
```

Window Size: Columns

Usage: Type the number of columns for the application's terminal window in the field.

Character application objects have this attribute:

Description

This attribute defines the number of columns in the terminal window, in the range 5–132.

Command Line

Command option: `--cols cols`

Usage: Replace *cols* with the number of columns in the terminal window.

The following example uses an 80-column window for the application.

```
--cols 80
```

Window Size: Height

Usage: Type the height of the application, in pixels, in the field.

The following objects have this attribute:

- Character application
- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute defines the height of the application, in pixels. The minimum height is 10 pixels, the maximum 65535 pixels.

Command Line

Command option: `--height pixels`

Usage: Replace *pixels* with the height of the application, in pixels. You must specify the height, even if this attribute is not required, for example because the application is configured with a [Window Type](#) setting of Client Window Management, or to display at the [Window Size: Client's Maximum Size](#).

The following example uses a 600-pixel high window to display the application.

```
--height 600
```

Window Size: Lines

Usage: Type the number of lines for the application's terminal window in the field.

Character application objects have this attribute:

Description

This attribute defines the number of lines in the terminal window, in the range 5-100.

Command Line

Command option: `-lines lines`

Usage: Replace *lines* with the number of lines in the terminal window.

The following example uses a 25-line window for the application.

```
--lines 25
```

Window Size: Maximized

Usage: Select or deselect the Maximized check box.

The following objects have this attribute:

- 3270 application
- 5250 application

Description

Specifies whether the emulator window is maximized.

These commands cause the window to be displayed at the maximum size possible when the TeemTalk for Unix emulator is loaded. The window retains the default number of lines and columns and includes all window elements, such as the title bar and soft buttons, if enabled.

Command Line

Command option: `--3270ma true|false`

Command option: `--ma true|false`

Usage: Specify `true` or `false`.

In the following example, the emulator window for a 3270 application is maximized.

```
--3270ma true
```

In the following example, the emulator window for a 5250 application is maximized.

```
--ma true
```

Window Size: Scale to Fit Window

Usage: Select or deselect the Scale to Fit Window check box.

The following objects have this attribute:

- 3270 application
- 5250 application
- Windows application
- X application

Description

This attribute specifies that the application is scaled to fit the window in which it is displayed.

This attribute is only available if the application has a [Window Type](#) setting of Independent Window or Kiosk.

If this attribute is selected, the application is always scaled to fit the window in which it is displayed. If you resize the window, SGD rescales the application to fit the new window size and scroll bars never display.

You can toggle between a scaled and an unscaled application by pressing the Scroll Lock key.

Command Line

Command option: `--scalable true | false`

Usage: Specify true or false.

The following example scales the application to fit the window.

```
--scalable true
```

Window Size: Width

Usage: Type the width of the application, in pixels, in the field.

The following objects have this attribute:

- Character application
- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute defines the width of the application, in pixels. The minimum width is 10 pixels, the maximum 65535 pixels.

Command Line

Command option: `--width pixels`

Usage: Replace *pixels* with the width of the application, in pixels. You must specify the width, even if this attribute is not required, for example because the application is configured with a [Window Type](#) setting of Client Window Management or to display at the [Window Size: Client's Maximum Size](#).

The following example uses a 300-pixel wide window to display the application.

```
--width 300
```

Window Type

Usage: Select a setting from the list.

The following objects have this attribute:

- Character application
- Windows application
- X application
- 3270 application
- 5250 application

Description

This attribute determines how the application is displayed to the user.

Some settings affect other attributes. For example, in the Administration Console, choosing Client Window Management disables the attributes for configuring the application's size. You can specify these attributes on the command line, but they have no effect.

The Administration Console options and their command line equivalents are shown in the following table.

| Administration Console | Command Line | Applies To | Description |
|--------------------------|--------------------------|-----------------------|---|
| Client Window Management | <code>clientwm</code> | X applications | <p>The application's windows behave in the same way as those of applications running on the client device. For example, the windows can be resized, moved, minimized and maximized using the client's normal window management controls.</p> <p>The object's Window Close Action attribute determines what happens when the user closes the application's last or main window.</p> <p>When listing object attributes on the command line, this attribute value is displayed as <code>multiplewindows</code>.</p> <p>Use for applications with many top-level resizable windows.</p> |
| Independent Window | <code>independent</code> | All application types | <p>The application appears in a new window, without any web browser toolbars or menus.</p> <p>This window can be resized, but this does not resize the application: the window includes scrollbars. The object's Window Size: Width and Window Size: Height attributes determine the size of the application.</p> <p>Closing the window ends or suspends the application session, depending on the object's Window Close Action attribute. A dialog is shown when the window is closed, asking you to confirm closure of the application.</p> <p>When listing object attributes on the command line, this attribute value is displayed as <code>awtwindow</code>.</p> |

| Administration Console | Command Line | Applies To | Description |
|------------------------|--------------|----------------------------|---|
| Kiosk | kiosk | All application types | <p>The application appears full-screen, with no window decoration.</p> <p>Users cannot resize or move the window.</p> <p>A pull-down header enables users to minimize or close the window.</p> <p>Use for full-screen desktop sessions.</p> |
| Local X Server | localx | X and Windows applications | <p>The application is displayed using an X server installed on the client device, if one is available. Otherwise, an independent window is used.</p> <p>Applications configured with this setting are <i>not resumable</i>, even if an independent window is used.</p> <p>The client device X server's host access control must grant access to the application server. See your X server's documentation for information about host access control.</p> |
| Seamless Window | seamless | Windows applications | <p>The application's windows behave like an application running on a Windows application server.</p> <p>If an application is started in a seamless window, you can toggle between a seamless and independent window by pressing the Scroll Lock key. This does not work if the SGD Client is in Integrated Mode.</p> <p>When listing object attributes on the command line, this attribute value is displayed as <code>seamlesswindows</code>.</p> <p>Do not use for full-screen desktop sessions. Use a kiosk or independent window instead.</p> |

Command Line

Command option: `--displayusing clientwm | independent | kiosk | localx | seamless`

Usage: Specify one of the valid settings. Not all settings are available for all types of application.

The following example displays the application as a full-screen desktop session.

```
--displayusing kiosk
```

The following example displays the application in an independent window.

```
--displayusing independent
```

Window Type: New Browser Window

Usage: Select or deselect the check box.

Document objects have this attribute.

Description

For users logged in to SGD using a browser, if this attribute is selected the URL specified for the object is displayed in a new browser window. If this attribute is not selected, the URL is displayed on the webtop.

Command Line

Command option: `--newbrowser true | false`

Usage: Specify true or false.

The following example displays the document in a new browser window.

```
--newbrowser true
```

Working Directory

Usage: Type the full path name of the directory in the field.

Windows application objects have this attribute.

Description

This attribute specifies the working directory to be used by the application.

You can use a backslash (\) or a forward slash (/) between subdirectories. On the command line, you might need to escape backslashes. For example, \\.

Command Line

Command option: `--workingdir dirname`

Usage: Replace *dirname* with the full path name of the working directory.

The following example specifies a working directory.

```
--workingdir "C:/Program Files/Indigo Insurance/data"
```

X Security Extension

Usage: Select or deselect the check box.

X application objects have this attribute.

Description

Whether to enable the X Security Extension for the application.

The X Security Extension divides X clients, also known as hosts, into trusted and untrusted clients. Untrusted clients cannot interact with windows and resources owned by trusted clients.

If you need to run an X application from an application server that might not be secure, enable the X Security Extension and run the application in untrusted mode. This restricts the operations that the X application can perform in the X server and protects the display.

To run an application in untrusted mode, do the following:

1. Configure the X application to use ssh as the [Connection Method](#).
2. Configure ssh to allow X11 forwarding.

The X Security Extension only works with versions of ssh that support the `-Y` option.

Command Line

Command option: `--securityextension true | false`

Usage: Specify true or false.

The following example enables the X Security Extension for the application.

```
--securityextension true
```

Commands

Oracle Secure Global Desktop (SGD) includes a built-in command set for controlling and configuring SGD. This chapter describes the available SGD commands and includes usage examples for each of the commands.

This chapter includes the following topics:

- [“The tarantella Command” on page 672](#)
- [“The tarantella archive Command” on page 675](#)
- [“The tarantella array Command” on page 675](#)
- [“The tarantella cache Command” on page 686](#)
- [“The tarantella config Command” on page 688](#)
- [“The tarantella emulatorsession Command” on page 692](#)
- [“The tarantella help Command” on page 699](#)
- [“The tarantella object Command” on page 700](#)
- [“The tarantella passcache Command” on page 762](#)
- [“The tarantella print Command” on page 770](#)
- [“The tarantella query Command” on page 780](#)
- [“The tarantella restart Command” on page 787](#)
- [“The tarantella role Command” on page 791](#)
- [“The tarantella security Command” on page 799](#)
- [“The tarantella service Command” on page 816](#)
- [“The tarantella setup Command” on page 826](#)
- [“The tarantella start Command” on page 827](#)
- [“The tarantella status Command” on page 830](#)
- [“The tarantella stop Command” on page 832](#)
- [“The tarantella tokencache Command” on page 836](#)
- [“The tarantella tscal Command” on page 839](#)

- “The `tarantella uninstall` Command” on page 844
- “The `tarantella version` Command” on page 845
- “The `tarantella webserver` Command” on page 846
- “The `tarantella webtopsession` Command” on page 850

The `tarantella` Command

You can control SGD from the command line using the `/opt/tarantella/bin/tarantella` command.

Syntax

| |
|--|
| <code>tarantella option [option-specific-arguments]</code> |
|--|

Description

Do not try to control the SGD server by running binaries directly, or by using `kill`. Using the `tarantella` command is the only supported way of controlling the SGD server.

The options for this command enable you to control the SGD server in different ways, or produce information about the SGD server. The `tarantella` command can be used in your own shell scripts to help automate your administration of SGD.

If the SGD server is running, most `tarantella` options can be run by root or *any user* in the `ttaserv` group. The `ttaserv` group does not have to be the user’s primary or effective group. See the table below for details of which users can use the command options.

If the SGD server is stopped, only root can use the `tarantella` command.

The following table shows the available options for this command.

| Option | Description | Can Be Run By | More Information |
|-----------------|---|-----------------------|--|
| archive | Archives the SGD server's log files. | root | "The tarantella archive Command" on page 675 |
| array | Creates and manages arrays of SGD servers. | SGD Administrators | "The tarantella array Command" on page 675 |
| cache | Manages the cache of Lightweight Directory Access Protocol (LDAP) data. | SGD Administrators | "The tarantella cache Command" on page 686 |
| config | Edits global and server-specific configuration. | root or ttaserv group | "The tarantella config Command" on page 688 |
| emulatorsession | Lists and controls application sessions. | root or ttaserv group | "The tarantella emulatorsession Command" on page 692 |
| help | Shows a list of SGD commands. | root or ttaserv group | "The tarantella help Command" on page 699 |
| object | Manipulates objects in the organizational hierarchy. | root or ttaserv group | "The tarantella object Command" on page 700 |
| passcache | Manipulates the password cache. | root or ttaserv group | "The tarantella passcache Command" on page 762 |
| print | Controls SGD printing services. | root or ttaserv group | "The tarantella print Command" on page 770 |
| query | Examines the SGD server's log files. | root | "The tarantella query Command" on page 780 |
| restart | Restarts SGD services. | root | "The tarantella restart Command" on page 787 |
| role | Gives users specific roles, and assigns applications specific to that role. | root or ttaserv group | "The tarantella role Command" on page 791 |
| security | Controls security services, manages certificates. | root | "The tarantella security Command" on page 799 |
| service | Manages service objects for LDAP and Active Directory authentication. | root or ttaserv group | "The tarantella service Command" on page 816 |
| setup | Changes Setup options, restores original objects. | root | "The tarantella setup Command" on page 826 |
| start | Starts SGD services. | root | "The tarantella start Command" on page 827 |

| Option | Description | Can Be Run By | More Information |
|---------------|--|-----------------------|--|
| status | Shows the current status of SGD servers in the array. | root or ttaserv group | “The tarantella status Command” on page 830 |
| stop | Stops SGD services. | root | “The tarantella stop Command” on page 832 |
| tokencache | Manipulates the token cache. | root or ttaserv group | “The tarantella tokencache Command” on page 836 |
| tscal | Manages Microsoft Windows TerminalServicesClient Access Licenses (CALs) for non-Windows clients. | root or ttaserv group | “The tarantella tscal Command” on page 839 |
| uninstall | Uninstalls SGD. | root | “The tarantella uninstall Command” on page 844 |
| version | Displays versions of installed SGD packages. | root or ttaserv group | “The tarantella version Command” on page 845 |
| webserver | Configures trusted users for the third-party authentication mechanism. | root | “The tarantella webserver Command” on page 846 |
| webtopsession | Lists and controls user sessions. | root or ttaserv group | “The tarantella webtopsession Command” on page 850 |

Note – All commands include a `--help` option. You can use `tarantella command --help` to get help on a specific command.

Examples

The following example stops and then restarts the SGD server, without displaying any messages.

```
# tarantella restart sgd --quiet
```

The following example adds a link for the Write-o-Win application to the assigned applications for members of the Global Administrators role.

```
$ tarantella role add_link --role global \
--link "o=applications/cn=Write-o-Win"
```

The `tarantella archive` Command

Archives the SGD server's log files.

Syntax

```
tarantella archive
```

Description

Archiving the logs compresses the files and moves them to a numbered subdirectory of the `/opt/tarantella/var/log` directory. A file `summary.txt` in this directory contains the results of performing the `tarantella query` command at the time of the archive.

Examples

The following example archives the SGD server's log files.

```
# tarantella archive
```

The `tarantella array` Command

This command enables SGD Administrators to set up and dismantle arrays of SGD servers.

The command can be run on any SGD server in the array.

Syntax

| |
|---|
| tarantella array add_backup_primary clean detach edit_backup_primary join list list_backup primaries make_primary remove_backup primaries |
|---|

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|-----------------------|---|--|
| add_backup_primary | Adds a secondary server to the backup primaries list for an array. | “tarantella array add_backup_primary” on page 677 |
| clean | Returns an SGD server to a preconfigured array state. | “tarantella array clean” on page 678 |
| detach | Removes secondary servers from an array. | “tarantella array detach” on page 679 |
| edit_backup_primary | Sets the position of a secondary server in the backup primaries list for an array. | “tarantella array edit_backup_primary” on page 680 |
| join | Adds a server to an array. | “tarantella array join” on page 681 |
| list | Lists the members of the array, identifying the primary server. | “tarantella array list” on page 683 |
| list_backup primaries | Displays the backup primaries list for an array. | “tarantella array list_backup primaries” on page 683 |
| make_primary | Makes a secondary server the primary server for the array that it is currently a member of. | “tarantella array make_primary” on page 684 |
| remove_backup_primary | Removes one or all secondary servers from the backup primaries list for an array. | “tarantella array remove_backup_primary” on page 685 |

Note – All commands include a `--help` option. You can use `tarantella array command --help` to get help on a specific command.

Examples

The following example adds the server `boston` to the array with primary server `newyork`.

```
$ tarantella array join --primary newyork.example.com \  
--secondary boston.example.com
```

The following example makes the secondary server `boston` the primary server in the array. The previous primary server becomes a secondary server.

```
$ tarantella array make_primary \  
--secondary boston.example.com
```

tarantella array add_backup_primary

Adds a secondary server to the list of backup primaries for an SGD array.

Syntax

```
tarantella array add_backup_primary  
                                --secondary serv  
                                --position [ first | last | position ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------------------------|---|
| <code>--secondary</code> | Specifies the peer Domain Name System (DNS) name of a secondary server to add to the backup primaries list. The server name must be the name of a secondary server in the array. You can only add one server at a time. |
| <code>--position</code> | The position of the secondary server in the backup primaries list. For <i>position</i> , you can either type a number, or you can use the <i>first</i> or <i>last</i> keyword to specify the first or last position in the list. The number 0 means first position in the list. |

Examples

The following example adds the secondary server boston to the backup primaries list for the array. The server is added in last position of the list.

```
$ tarantella array add_backup_primary \  
--secondary boston.example.com --position last
```

tarantella array clean

Deletes array information and returns an SGD server to a preconfigured array state. By default, a confirmation prompt is displayed.

Note – This command only affects the SGD server where the command is run.

Syntax

```
tarantella array clean [ --standalone | --contactmembers ]  
                        [ --quiet ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|------------------|--|
| --standalone | Deletes all array information and returns the SGD server to standalone state. In standalone state, the server is not a member of an array. This is the default option. Caution - After you run this command on the primary server in an SGD array, the secondary servers will not be able to contact the primary server. |
| --contactmembers | The SGD server remains in an array with any servers that are contactable and that report the same array membership. Non-contactable servers, or servers that disagree on the current array membership, are not included in the array. |
| --quiet | Does not display a confirmation prompt before cleaning. |

Note – After running this command, it is advisable to run the `tarantella status` command on the SGD server to check the status of the server.

Examples

The following example removes array configuration and returns the SGD server where the command is run to a standalone state. A confirmation prompt is not shown.

```
$ tarantella array clean --quiet
```

tarantella array detach

Removes a secondary server from the array of SGD servers it belongs to.

Syntax

```
tarantella array detach --secondary serv
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------------------------|---|
| <code>--secondary</code> | Specifies the peer DNS name of a secondary server to remove. The server name must be the name of a secondary server in the same array. You can only remove one server at a time. |

To remove the primary server from an array, first use `tarantella array make_primary` to make another server the primary server and then detach the old primary server.

When you use this command, the secondary server is removed from the backup primaries list for the SGD array.

When you remove a server from an array, it loses its license keys.

Note – After running this command, it is advisable to wait until SGD has copied the changes to all SGD servers in the array before running any further `tarantella array` commands. Run the `tarantella status` command on the primary SGD server to check the status of the array.

If you are using secure intra-array communication, the secondary server generates its own Certificate Authority (CA) certificate and its own server peer Secure Sockets Layer (SSL) certificate when it is detached.

Examples

The following example removes the secondary server `boston` from the array.

```
$ tarantella array detach --secondary boston.example.com
```

`tarantella array edit_backup_primary`

Sets the position of a secondary server in the list of backup primaries for an SGD array.

Syntax

```
tarantella array edit_backup_primary
--secondary serv
--position [ first | last | position ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|-------------|---|
| --secondary | Specifies the peer DNS name of the secondary server in the backup primaries list. The server name must be the name of a secondary server in the array. You can only change the position for one server at a time. |
| --position | The position of the secondary server in the backup primaries list. For <i>position</i> , you can either type a number, or you can use the <i>first</i> or <i>last</i> keyword to specify the first or last position in the list. The number 0 means first position in the list. |

Examples

The following example sets the position of the secondary server *boston* in the backup primaries list for the array. The server is moved to last position in the list.

```
$ tarantella array edit_backup_primary \  
--secondary boston.example.com --position last
```

tarantella array join

Adds a server to an array of SGD servers, either as a primary or a secondary server.

Syntax

```
tarantella array join [ --primary pserv ]  
                      [ --secondary sserv ]
```

Description

The clock on the server joining the array must be in synchronization with the clocks on the other servers in the array. If the time difference is more than one minute, this command will fail.

After running this command, it is advisable to wait until SGD has copied the changes to all SGD servers in the array before running any further `tarantella array` commands. Run the `tarantella status` command on the primary SGD server to check the status of the array.

It is best to use a fully-qualified DNS name when specifying the server to add to an array.

The following table shows the available options for this command.

| Option | Description |
|--------------------------|--|
| <code>--primary</code> | Specifies the peer DNS name of the primary server in the array. Defaults to the server where the command is run. |
| <code>--secondary</code> | Specifies the peer DNS name of the server to add. The secondary server must be the only member of an array. Defaults to the server where the command is run. You can only add one secondary server at a time. |

When you use this command, the secondary server is added to the end of the backup primaries list for the SGD array.

If the server you add has been load balancing application servers using Advanced Load Management, use the `tarantella restart sgd --warm` command to do a warm restart of the new server after it has joined the array. If the array to which the new server is joined is using Advanced Load Management, do a warm restart of the whole array after the new server has joined.

If you are using secure intra-array communication, you are prompted to accept the CA certificate of either the primary server or the secondary server, depending on where you ran the command.

Examples

The following example adds the server boston to the array with newyork as its primary server.

```
$ tarantella array join \  
--primary newyork.example.com \  
--secondary boston.example.com
```

The following example adds the server where the command is run to the array with newyork as its primary server.

```
$ tarantella array join \  
--primary newyork.example.com
```

tarantella array list

Lists each member of the array of SGD servers, identifying the primary server.

Syntax

```
tarantella array list
```

Examples

The following example lists all SGD servers in the array.

```
$ tarantella array list
```

tarantella array list_backup primaries

Displays the contents of the backup primaries list for an SGD array.

Syntax

```
tarantella array list_backup primaries [ --first | --last | --all ]
```

Description

If no options are specified, this command displays all entries in the backup primaries list.

The following table shows the available options for this command.

| Option | Description |
|---------|--|
| --first | Displays the first entry in the backup primaries list. |
| --last | Displays the last entry in the backup primaries list. |
| --all | Displays all entries in the backup primaries list. This is the default option. |

Examples

The following example displays all entries in the backup primaries list for the SGD array.

```
$ tarantella array list_backup primaries
```

tarantella array make_primary

Makes a secondary server the primary server for the array that it is currently a member of. The previous primary server becomes a secondary server.

Syntax

```
tarantella array make_primary --secondary serv
```

Description

The following table shows the available options for this command.

| Subcommand | Description |
|-------------|--|
| --secondary | Specifies the peer DNS name of the secondary server to be made the primary server. |

Note – After running this command, it is advisable to wait until SGD has copied the changes to all SGD servers in the array before running any further `tarantella array` commands. Run the `tarantella status` command on the primary SGD server to check the status of the array.

If you are using secure intra-array communication, the new primary becomes the certificate authority for the array and issues new server peer SSL certificates to all SGD servers in the array.

Examples

The following example makes the secondary server `boston` the primary server in the array.

```
$ tarantella array make_primary \  
--secondary boston.example.com
```

tarantella array remove_backup_primary

Removes one or all secondary servers from the list of backup primaries for an SGD array.

Syntax

```
tarantella array remove_backup_primary [ --secondary serv | --all ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|-------------|---|
| --secondary | Specifies the peer DNS name of the secondary server to remove from the backup primaries list. The server name must be the name of a secondary server in the array. With this option, you can only remove one server at a time from the list. |
| --all | Removes all secondary servers from the backup primaries list. |

Examples

The following example removes the secondary server boston from the backup primaries list for the array.

```
$ tarantella array remove_backup_primary \  
--secondary boston.example.com
```

The tarantella cache Command

Modifies the cache of directory services data.

Syntax

```
tarantella cache
[ --flush ldapgroups|ldapconn|ldapconn-lookups|krb5config|all ]
[ --refresh ldapgroups ]
[ --populate ldapgroups ]
```

Description

This command modifies the cache of directory services data used with the following:

- Active Directory authentication
- LDAP authentication
- Third-party authentication (if the LDAP search method is enabled)
- LDAP assignments (also known as Directory Services Integration)

The following table shows the available options for this command.

| Option | Description |
|------------|--------------------------------|
| --flush | Flushes the cache. |
| --refresh | Updates the data in the cache. |
| --populate | Adds data to the cache. |

The following table shows the available settings for the command options.

| Setting | Description |
|------------------|---|
| ldapgroups | LDAP group data. Used for LDAP assignments. |
| ldapconn | LDAP connections Used to reset all LDAP connections. |
| ldapconn-lookups | LDAP search data. Used for LDAP assignments. |
| krb5config | The Kerberos configuration settings. Used to update the Kerberos settings without restarting the SGD server. Used only for Active Directory authentication. |
| all | Flushes all LDAP data. |

The `--populate` option adds LDAP group and LDAP group membership information to the cache. SGD searches the local repository for objects with LDAP group assignments and adds the LDAP groups to the cache. SGD then queries the directory for the membership of each LDAP group and adds the list of users to the cache.

The `--refresh` option updates the cache with the current membership of LDAP groups. SGD searches the cache for LDAP groups. SGD then queries the directory for the membership of each LDAP group and adds the list of users to the cache.

Examples

The following example flushes the cache of all LDAP data.

```
$ tarantella cache --flush all
```

The `tarantella config` Command

The `tarantella config` command lists and configures global settings, and also server-specific settings for any SGD server in the array.

Syntax

```
tarantella config list | edit
```

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|------------|---|--|
| list | Lists global and server-specific attributes and their current values. | “tarantella config list” on page 691 |
| edit | Edits global and server-specific attributes. | “tarantella config edit” on page 689 |

Note – All commands include a `--help` option. You can use `tarantella config subcommand --help` to get help on a specific command.

Examples

The following example lists server-specific attributes from the server `newyork.indigo-insurance.com`.

```
$ tarantella config list --server newyork.indigo-insurance.com
```

The following example sets the `cpe-maxsessions` attribute to 10 for the server where the command is run.

```
$ tarantella config edit --cpe-maxsessions 10
```

tarantella config edit

Edits global and server-specific attributes.

Syntax

```
tarantella config edit { { --setting value... }...  
                        [ --array | --server serv... ]  
                        } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|---------------------------------|---|
| <code>--setting value...</code> | Names an attribute you want to edit, and its new value or values. |

| Option | Description |
|-----------------------|---|
| <code>--array</code> | When configuring a server-specific attribute, applies the change to all SGD servers in the array. |
| <code>--server</code> | When configuring a server-specific attribute, applies the change to each named <i>serv</i> in the array. Use a peer DNS name or IP address for each server. |
| <code>--file</code> | Specifies a file containing a batch of commands to edit attributes. |

If neither `--array` nor `--server` is specified, the command sets server-specific attributes for the SGD server where the command is run.

Use `tarantella config list` to see a list of *settings* you can change.

For detailed information on *global* attributes, see [Appendix A](#).

For detailed information on *server-specific* attributes, see [Appendix B](#).

Some attribute settings are lists. When editing a list on the command line, it is best to enclose each item in the list in quotes, and separate each item with a space. For example:

```
$ tarantella config edit --tarantella-config-tpeconfig-logfilter \
  "tpe/*/*" "pem/*/*"
```

Alternatively, you can put each item in the list on a separate line. For example:

```
$ tarantella config edit --tarantella-config-tpeconfig-logfilter \
  "tpe/*/*" \
  "pem/*/*"
```

If you omit the quotes, you must use a backslash (\) to escape any characters, such as asterisks (*), to prevent your shell from expanding them.

On the command line, when you display attribute settings that are lists, the setting is shown as a comma-separated list. For example:

```
$ tarantella config list --tarantella-config-tpeconfig-logfilter
tarantella-config-tpeconfig-logfilter: tpe/*/*,pem/*/*
```

Examples

The following example sets the `cpe-exitafter` attribute to 50 on SGD servers `newyork.indigo-insurance.com` and `boston.indigo-insurance.com`.

```
$ tarantella config edit --cpe-exitafter 50 \  
--server newyork.indigo-insurance.com boston.indigo-insurance.com
```

The following example sets the `cpe-maxsessions` attribute to 10 for the server where the command is run.

```
$ tarantella config edit --cpe-maxsessions 10
```

tarantella config list

Lists global and server-specific attributes and their current values.

Syntax

```
tarantella config list { [ --setting... ]  
                        [ --server serv ]  
                        } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|------------------------|--|
| <code>--setting</code> | Names an attribute you want to list the value of. If no <code>--setting</code> is specified, all global and server-specific attributes are listed. |
| <code>--server</code> | Lists server-specific attributes for the specified SGD server in the array. Use a peer DNS name or IP address. If omitted, lists server-specific attributes for the SGD server where the command is run. |
| <code>--file</code> | Specifies a file containing a batch of commands to list attributes. |

For detailed information on *global* attributes, see [Appendix A](#).

For detailed information on *server-specific* attributes, see [Appendix B](#).

Examples

The following example lists global attributes, and server-specific attributes for the server `newyork.indigo-insurance.com`.

```
$ tarantella config list --server newyork.indigo-insurance.com
```

The following example lists the value of the `array-port-unencrypted` attribute.

```
$ tarantella config list --array-port-unencrypted
```

The `tarantella emulatorsession` Command

This command enables SGD Administrators to list and manipulate application sessions.

Syntax

```
tarantella emulatorsession list | info | shadow | suspend | end
```

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|-------------------|---|--|
| <code>list</code> | Lists application sessions. | <code>"tarantella emulatorsession list"</code> on page 693 |
| <code>info</code> | Displays detailed information about application sessions. | <code>"tarantella emulatorsession info"</code> on page 695 |

| Subcommand | Description | More Information |
|------------|---------------------------------|--|
| shadow | Shadows an application session. | "tarantella emulatorsession shadow" on page 696 |
| suspend | Suspends application sessions. | "tarantella emulatorsession suspend" on page 697 |
| end | Ends application sessions. | "tarantella emulatorsession end" on page 698 |

Note – All commands include a `--help` option. You can use `tarantella emulatorsession subcommand --help` to get help on a specific command.

Examples

The following example lists Emma Rald's application sessions.

```
$ tarantella emulatorsession list \
--person "o=Indigo Insurance/cn=Emma Rald"
```

The following example shadows the application session with the specified session ID.

```
$ tarantella emulatorsession shadow \
"paris.indigo-insurance.com:965127448604:...%2f_ens%2fo=Indigo
Insurance%2fcn=Emma Rald"
```

tarantella emulatorsession list

Lists application sessions matching the criteria specified. Information shown includes session IDs, which are used with other `tarantella emulatorsession` commands.

An example session ID is `paris.indigo-insurance.com:965127448604:...%2f_ens%2fo=Indigo Insurance%2fcn=Emma Rald`.

Session IDs can contain spaces, so make sure you quote them.

Syntax

```
tarantella emulatorsession list [ --person pobj ]  
                                [ --application appobj ]  
                                [ --appserver hobj ]  
                                [ --server serv ]  
                                [ --format text|count|xml ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|---------------|---|
| --person | Lists application sessions matching the name of the person specified. Use the user identity of the person. |
| --application | Lists application sessions matching the name of application specified. Use the name of an application object in the local repository. |
| --appserver | Lists application sessions matching the name of application server specified. Use the name of an application server object in the local repository. |
| --server | Lists application sessions hosted by the SGD server specified. Use the name of an application server object in the local repository (if the SGD server has an application server object), or the peer DNS name for the server. |
| --full | Includes the current IP address of the client and the status of the application session in the output. It takes longer to display this information. |
| --format | Specifies the output format. The default setting is text. Use count to display only the number of matching sessions. |

If --person, --application, --appserver, and --server are all omitted, all application sessions are listed.

Examples

The following example lists Emma Rald's application sessions.

```
$ tarantella emulatorsession list \  
--person "o=Indigo Insurance/cn=Emma Rald"
```

The following example lists all application sessions hosted by the SGD server `boston.indigo-insurance.com`. This is the server on which the Protocol Engines run.

```
$ tarantella emulatorsession list \
--server boston.indigo-insurance.com
```

tarantella emulatorsession info

Displays detailed information about application sessions.

Syntax

```
tarantella emulatorsession info [ --sessid sessid... ]
                                [ --peid peid... ]
                                [ --format text|xml|quiet ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|-----------------------|--|
| <code>--sessid</code> | Displays detailed information on application sessions matching the session IDs listed. Use <code>tarantella emulatorsession list</code> to find out session IDs. |
| <code>--peid</code> | Displays detailed information on application sessions matching the Protocol Engine process IDs listed. Valid process IDs are as follows: <ul style="list-style-type: none">• A number, such as 3456, representing the process ID on the application server where the command is run• A combination of peer DNS name and process ID, for example <code>boston.indigo-insurance.com:3456</code>, representing the process ID on the SGD server named. |
| <code>--format</code> | Specifies the output format. The default setting is <code>text</code> . With <code>--format quiet</code> , no messages are displayed. |

The exit code indicates the number of session IDs and process IDs named that do not exist.

Examples

The following example displays detailed information on application sessions matching the Protocol Engine process IDs “3456” and “4567” on the application server where the command is run.

```
$ tarantella emulatorsession info --peid 3456 4567
```

tarantella emulatorsession shadow

Shadows an application session, enabling you and the user to interact with the application simultaneously. Only SGD Administrators can shadow application sessions. You can only shadow Windows and X applications. Suspended applications cannot be shadowed.

Syntax

```
tarantella emulatorsession shadow sessid
                                     [ --read-only ]
                                     [ --silent ]
                                     [ --format text|quiet ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|---------------|--|
| <i>sessid</i> | Shadows the application session with the specified session ID. Use tarantella emulatorsession list to find out session IDs. |
| --read-only | Enables an Administrator to shadow a session without being able to interact with the application. |
| --silent | Enables an Administrator to shadow a session and interact with the application. The user is <i>not notified</i> that an Administrator wants to shadow their session and they cannot refuse permission. If this is used with --read-only, the user does not know they are being shadowed and the Administrator cannot interact with the application. Note - In some countries, it is illegal to shadow a user without their knowledge. It is your responsibility to comply with the law. |
| --format | Specifies the output format. The default setting is <code>text</code> . With --format <code>quiet</code> , no messages are displayed. |

Note – You can also shadow a session from the Global Settings → Application Sessions tab of the Administration Console. You select the session from either the user profile object or the application object. However, using the Administration Console does not enable you to shadow a session in read-only mode or silent mode.

If `--silent` is not used, the user is notified that an Administrator wants to shadow their session and they can refuse permission. The user is also notified when shadowing ends.

The exit code is 0 for success, 1 if the session does not exist, 2 if the session is not shadowable, or 3 if the session is suspended.

Examples

The following example shadows the application session with the specified session ID.

```
$ tarantella emulatorsession shadow \  
"paris.indigo-insurance.com:965127448604:...%2f_ens%2fo=Indigo  
Insurance%2fcn=Emma Rald"
```

The following example shadows the application session with the specified session ID without the user knowing that they are being shadowed. The Administrator is unable to interact with the application.

```
$ tarantella emulatorsession shadow \  
"paris.indigo-insurance.com:965127448604:...%2f_ens%2fo=Indigo  
Insurance%2fcn=Emma Rald" \  
--read-only --silent
```

tarantella emulatorsession suspend

Suspends application sessions.

Syntax

```
tarantella emulatorsession suspend sessid...  
[ --format text|quiet ]
```

Description

The following table shows available options for this command.

| Option | Description |
|-------------------|--|
| <i>sessid</i> ... | Suspends the application sessions with the specified session IDs. Use tarantella emulatorsession list to find out session IDs. |
| --format | Specifies the output format. The default setting is text. With --format quiet, no messages are displayed. |

The exit code is 0 for success, 1 if some sessions do not exist, 2 if some sessions are already suspended, or 3 if there is a mixture of nonexistent and suspended sessions.

Examples

The following example suspends the application session with the specified session ID.

```
$ tarantella emulatorsession suspend \  
"paris.indigo-insurance.com:965127448604:...%2f_ens%2fo=Indigo  
Insurance%2fcn=Emma Rald"
```

tarantella emulatorsession end

Ends application sessions. The applications exit immediately, which might result in loss of data for users.

Syntax

```
tarantella emulatorsession end sessid...  
[ --format text|quiet ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|-------------------|---|
| <i>sessid</i> ... | Specifies the session IDs of the application sessions to end. Use <code>tarantella emulatorsession list</code> to find out session IDs. |
| --format | Specifies the output format. The default setting is text. With --format quiet, no messages are displayed. |

The exit code of the command is 0 if all sessions were successfully ended, or 1 if some session IDs did not exist.

Examples

The following example ends the specified application session.

```
$ tarantella emulatorsession end \  
"paris.indigo-insurance.com:965127448604:...%2f_ens%2fo=Indigo  
Insurance%2fcn=Emma Rald"
```



The tarantella help Command

Shows a list of the SGD commands.

Syntax

```
tarantella help
```

Description

Shows the list of SGD commands.

To get help on a particular command, use `tarantella command --help`.

Examples

The following example shows the list of SGD commands.

```
$ tarantella help
```

The `tarantella object` Command

The `tarantella object` command enables you to create, list, edit, and delete objects in the organizational hierarchy. You can also add and remove assigned applications links, configure application server load balancing for each application, and add and remove group members.

Syntax

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------|----------|--|----------|--|-------------|--|------------|--|--------|--|------|--|-----------------|--|---------------|--|-------------|--|-------------|--|-------------|--|---------------|--|--------|--|---------|--|----------------|--|-----------|--|----------|--|---------|--|-------------|--|------------|--|----------------|--|----------|--|-------------|--|-------------|--|----------------|--|---------------|--|--------|--|--------|
| tarantella object | add_host | | add_link | | add_mapping | | add_member | | delete | | edit | | list_attributes | | list_contents | | new_3270app | | new_5250app | | new_charapp | | new_container | | new_dc | | new_doc | | new_dynamicapp | | new_group | | new_host | | new_org | | new_orgunit | | new_person | | new_windowsapp | | new_xapp | | remove_host | | remove_link | | remove_mapping | | remove_member | | rename | | script |
|-------------------|----------|--|----------|--|-------------|--|------------|--|--------|--|------|--|-----------------|--|---------------|--|-------------|--|-------------|--|-------------|--|---------------|--|--------|--|---------|--|----------------|--|-----------|--|----------|--|---------|--|-------------|--|------------|--|----------------|--|----------|--|-------------|--|-------------|--|----------------|--|---------------|--|--------|--|--------|

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|-----------------|--|---|
| add_host | Adds application servers to the list of those that can run an application. | “tarantella object add_host” on page 703 |
| add_link | Adds assigned applications links. | “tarantella object add_link” on page 704 |
| add_mapping | Adds mappings to a dynamic application. | “tarantella object add_mapping” on page 705 |
| add_member | Adds members to a group. | “tarantella object add_member” on page 707 |
| delete | Permanently deletes objects from the organizational hierarchy. | “tarantella object delete” on page 708 |
| edit | Edits attributes for an object. | “tarantella object edit” on page 709 |
| list_attributes | Lists attributes of an object. | “tarantella object list_attributes” on page 710 |
| list_contents | Lists the contents of an OU or an organization. | “tarantella object list_contents” on page 711 |
| new_3270app | Creates 3270 application objects. | “tarantella object new_3270app” on page 712 |
| new_5250app | Creates 5250 application objects. | “tarantella object new_5250app” on page 717 |
| new_charapp | Creates character application objects. | “tarantella object new_charapp” on page 722 |
| new_container | Creates Active Directory container objects. | “tarantella object new_container” on page 727 |

| Subcommand | Description | More Information |
|-----------------------------|---|--|
| <code>new_dc</code> | Creates domain component objects. | “tarantella object new_dc” on page 727 |
| <code>new_doc</code> | Creates document objects. | “tarantella object new_doc” on page 728 |
| <code>new_dynamicapp</code> | Creates dynamic application objects. | “tarantella object new_dynamicapp” on page 730 |
| <code>new_group</code> | Creates group objects. | “tarantella object new_group” on page 731 |
| <code>new_host</code> | Creates application server objects. | “tarantella object new_host” on page 733 |
| <code>new_org</code> | Creates organization objects. | “tarantella object new_org” on page 736 |
| <code>new_orgunit</code> | Creates organizational unit objects. | “tarantella object new_orgunit” on page 738 |
| <code>new_person</code> | Creates user profile objects. | “tarantella object new_person” on page 741 |
| <code>new_windowsapp</code> | Creates Windows application objects. | “tarantella object new_windowsapp” on page 744 |
| <code>new_xapp</code> | Creates X application objects. | “tarantella object new_xapp” on page 750 |
| <code>remove_host</code> | Removes application servers from those that can run an application. | “tarantella object remove_host” on page 755 |
| <code>remove_link</code> | Removes assigned applications links. | “tarantella object remove_link” on page 756 |
| <code>remove_mapping</code> | Removes mappings for a dynamic application. | “tarantella object remove_mapping” on page 757 |
| <code>remove_member</code> | Removes members from groups. | “tarantella object remove_member” on page 759 |
| <code>rename</code> | Renames or moves an object. | “tarantella object rename” on page 760 |
| <code>script</code> | Runs a batch script of object commands. | “tarantella object script” on page 761 |

Note – All commands include a `--help` option. You can use `tarantella object subcommand --help` to get help on a specific command.

Examples

The following example lists the objects that belong to the organizational unit Sales.

```
$ tarantella object list_contents \  
--name "o=Indigo Insurance/ou=Sales"
```

tarantella object add_host

Adds application servers to the list of those that can run an application, for application server load balancing.

Syntax

```
tarantella object add_host { --name obj...  
                             --host hobj...  
                             } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------|---|
| --name | The names of application objects you want to configure load balancing for. |
| --host | The names of application server objects you want to add to the load balancing pool. |
| --file | A file containing a batch of commands to configure application server load balancing. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example adds the application server rome to the load balancing pool for the application Slide-o-Win.

```
$ tarantella object add_host \  
--name "o=applications/cn=Slide-o-Win" \  
--host "o=appservers/ou=Sales/cn=rome"
```

The following example adds the group WinHosts to the load balancing pool for the applications Write-o-Win and Slide-o-Win. Load balancing is performed across all the application servers in WinHosts.

```
$ tarantella object add_host \  
--name "o=applications/cn=Write-o-Win" \  
"o=applications/cn=Slide-o-Win" \  
--host "o=appservers/cn=WinHosts"
```

tarantella object add_link

Adds assigned applications links for an object.

Syntax

```
tarantella object add_link { --name obj...  
                             --link lobj...  
                             } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------|---|
| --name | The names of objects you want to add assigned applications links for. |
| --link | The names of assigned applications links you want to add. |
| --file | A file containing a batch of commands to add assigned applications links. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example adds the Write-o-Win application to Violet Carson's assigned applications.

```
$ tarantella object add_link \  
--name "o=Indigo Insurance/ou=Sales/cn=Violet Carson" \  
--link "o=applications/cn=Write-o-Win"
```

The following example adds the group Applications to the assigned applications of the organizational units Sales and Marketing. Everyone who inherits assigned applications from one of these OUs, for example, they belong to that OU and [Inherit Assigned Applications from Parent](#) is selected for their user profile object, sees all the applications in the group in their assigned applications.

```
$ tarantella object add_link \  
--name "o=Indigo Insurance/ou=Sales" \  
--name "o=Indigo Insurance/ou=Marketing" \  
--link "o=applications/cn=Applications"
```

tarantella object add_mapping

Adds mappings for a dynamic application object. See [“Dynamic Application Object” on page 539](#).

Syntax

```
tarantella object add_mapping { --name obj
                                --mapping [type=application]
                                } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description | More Information |
|-----------|---|--|
| --name | The name of a dynamic application object. | “Name” on page 628 |
| --mapping | A mapping between a type and the name of an application object. | “Mappings Tab” on page 621 |
| --file | A file containing a batch of commands to add mappings. | |

When you add a mapping, the applications that are mapped must already exist.

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example adds a windows type mapping to the winApp dynamic application object.

```
$ tarantella object add_mapping \
  --name "o=applications/cn=winApp" \
  --mapping windows="o=applications/cn=windows_desktop"
```

The following example adds a solaris type mapping to the desktopApp dynamic application object.

```
$ tarantella object add_mapping \
  --name "o=applications/cn=desktopApp" \
  --mapping solaris="o=applications/cn=solaris_desktop"
```

tarantella object add_member

Adds objects to groups.

Syntax

```
tarantella object add_member { --name obj...
                               --member mobj...
                               } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------|---|
| --name | Specifies the names of group objects you want to add members for. |
| --member | Specifies the names of objects you want to add to the groups. |
| --file | Specifies a file containing a batch of commands to add group members. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example adds the Write-o-Win application to the group Applications.

```
$ tarantella object add_member \
  --name "o=applications/cn=Applications" \
  --member "o=applications/cn=Write-o-Win"
```

The following example adds the three application server objects rome, brussels, and berlin to the group WinHosts. This group can be added to an application’s [Hosting Application Servers Tab](#) to perform load balancing between the application servers. From the command line, use `tarantella object add_host`.

```
$ tarantella object add_member \
  --name "o=appservers/cn=WinHosts" \
```

```
--member "o=appservers/ou=Sales/cn=rome" \  
--member "o=appservers/cn=brussels" \  
--member "o=appservers/ou=Marketing/cn=berlin"
```

tarantella object delete

Permanently deletes objects from the organizational hierarchy.

Syntax

```
tarantella object delete { --name obj [ --children ] } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|------------|--|
| --name | Specifies the name of the object you want to delete. |
| --children | When deleting organizational units, Active Directory containers or domain components, confirms that you want to delete the object and all objects that belong to it, recursively. As a safeguard, it is impossible to delete an organizational unit, Active Directory container or domain component without specifying --children. |
| --file | Specifies a file containing a batch of commands to delete objects. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example removes the user profile object for Violet Carson.

```
$ tarantella object delete \  
--name "o=Indigo Insurance/ou=Sales/cn=Violet Carson"
```

The following example deletes the organizational unit Sales.

```
$ tarantella object delete \  
--name "o=Indigo Insurance/ou=Sales" \  
--children
```

tarantella object edit

Edits the attributes of an object in the organizational hierarchy.

Syntax

```
tarantella object edit { --name obj  
                        { --attribute [value] }...  
                        } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|------------------------------------|---|
| --name | Specifies the name of the object you want to edit the attributes of. |
| {--attribute [<i>value</i>] }... | Specifies the attribute names you want to edit, and their new values. The valid <i>attributes</i> depend on the type of object. See the <code>tarantella object new_object_type</code> documentation for the appropriate list. For example, when editing attributes for an application object you can specify <code>--display</code> using to edit the Window Type attribute. If you omit <i>value</i> for an attribute, it is deleted from the object. |
| --file | Specifies a file containing a batch of commands to edit attributes. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example changes the [Inherit Assigned Applications from Parent](#) attribute for the organizational unit Sales.

```
$ tarantella object edit \  
  --name "o=Indigo Insurance/ou=Sales" \  
  --inherit false
```

tarantella object list_attributes

Lists the attributes of an object in the organizational hierarchy.

Syntax

```
tarantella object list_attributes { --name obj  
                                   [ --attribute... ]  
                                   } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------------------------|--|
| --name | Specifies the name of the object you want to list the attributes of. |
| {--attribute [value]}... | Specifies the attribute names you want to list. The valid <i>attributes</i> depend on the type of object. See the <code>tarantella object new_object_type</code> documentation for the appropriate list. For example, when listing attributes for an application object you can specify <code>--displayusing</code> to edit the Window Type attribute. |
| --file | Specifies a file containing a batch of commands to list attributes. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example lists all attributes for the Sales organizational unit.

```
$ tarantella object list_attributes \  
--name "o=Indigo Insurance/ou=Sales"
```

The following example lists the [Email Address](#) and [Login](#) attributes for the user profile object for Rusty Spanner.

```
$ tarantella object list_attributes \  
--name "o=Indigo Insurance/ou=IT/cn=Rusty Spanner" \  
--email --enabled
```

tarantella object list_contents

Lists the objects that belong to a particular object in the organizational hierarchy.

Syntax

```
tarantella object list_contents { --name obj } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------|--|
| --name | Specifies the name of the object you want to list the contents of. |
| --file | Specifies a file containing a batch of commands to list object contents. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example lists all the objects within the organizational unit Sales.

```
$ tarantella object list_contents \  
--name "o=Indigo Insurance/ou=Sales"
```

tarantella object new_3270app

Creates one or more 3270 application objects. See [“3270 Application Object” on page 526](#).

Syntax

```
tarantella object new_3270app {
  --name obj
  --width pixels
  --height pixels
  [ --description text ]
  [ --args args ]
  [ --method rexec|telnet|ssh ]
  [ --resumable never|session|always ]
  [ --endswhen lastclient|windowmanager|windowmanageralone|
  [ --maxinstances 0|instances ]
  [ --displayusing clientwm|independent|kiosk|localx ]
  [ --maximize true|false ]
  [ --scalable true|false ]
  [ --icon icon_name ]
  [ --hints hint...]
  [ --hostname host ]
  [ --portnumber tcp ]
  [ --3270tnclose 0|1|2|3 ]
  [ --3270kt pc|sun4|sun5|hp ]
  [ --3270bl 0|1|2|3|4 ]
  [ --3270ma true|false ]
  [ --3270mb true|false ]
  [ --3270si true|false ]
  [ --3270fg color ]
  [ --3270bg color ]
  [ --roottype default|custom ]
  [ --rootcolor color ]
  [ --compression automatic|on|off ]
  [ --execution automatic|inorder|optimized ]
  [ --interlaced automatic|on|off ]
  [ --accel true|false ]
  [ --delayed true|false ]
  [ --ldapusers user_dn... ]
  [ --ldapgroups group_dn... ]
  [ --ldapsearch search_string... ]
  [ --env setting... ]
  [ --login script ]
  [ --winmgr command... ]
  [ --resumetimeout mins ]
  [ --middlemouse ms ]
  [ --windowclose notifyapp|killapp|suspendsession|endsession ]
  [ --euro unicode|iso8859-15 ]
  [ --dpi monitordpi ]
  [ --keepopen true|false ]
  [ --lockkeymap true|false ]
  [ --ssharguments args ]
```

```
[ --share true|false ]
} | --file file
```

Description

SGD uses the third-party TeemTalk for Unix emulator for 3270 applications. See the TeemTalk for Unix User's Guide supplied with SGD for details.

The following table shows the available options for this command.

| Option | Description | More Information |
|----------------|--|--|
| --name | The common name of the object in the SGD datastore. | "Name" on page 628 |
| --width | The width of the application, in pixels. | "Window Size: Width" on page 664 |
| --height | The height of the application, in pixels. | "Window Size: Height" on page 661 |
| --description | A text description of the object. | "Comment" on page 579 |
| --args | The command-line arguments to use when starting the application. | "Arguments for Command" on page 555 |
| --method | The mechanism used by the SGD server to access the application server and start the application. | "Connection Method" on page 581 |
| --resumable | Resumability behavior for the application. | "Application Resumability" on page 549 |
| --endswhen | When the application session ends. | "Session Termination" on page 640 |
| --maxinstances | The maximum number of instances of the application a user can run simultaneously. | "Number of Sessions" on page 630 |
| --displayusing | How the application is displayed to the user. | "Window Type" on page 665 |
| --maximize | The initial size of the application. | "Window Size: Client's Maximum Size" on page 660 |
| --scalable | Scale the application to fit the window in which it is displayed. | "Window Size: Scale to Fit Window" on page 663 |
| --icon | Webtop icon for the application. | "Icon" on page 606 |
| --hints | String containing additional name-value data for the application. | "Hints" on page 602 |
| --hostname | The 3270 host that runs the application. | "Server Address" on page 639 |
| --portnumber | The Transmission Control Protocol (TCP) port number used to connect to the 3270 host. | "Server Port" on page 639 |

| Option | Description | More Information |
|---------------|--|---|
| --3270tnclose | Behavior when telnet connection to the 3270 host is closed. | “Connection Closed Action” on page 580 |
| --3270kt | Layout to use for mapping the keyboard to the terminal being emulated. | “Keyboard Type” on page 612 |
| --3270bl | Number of “soft button” levels to display. | “Displayed Soft Buttons” on page 591 |
| --3270ma | Maximizes the emulator window. | “Window Size: Maximized” on page 662 |
| --3270mb | Enables the application’s menu bar. | “Menu Bar” on page 625 |
| --3270si | Enables the File and Settings menu items. | “‘File’ and ‘Settings’ Menus” on page 596 |
| --3270fg | Text color in the application’s text window. | “Foreground Color” on page 599 |
| --3270bg | Background color of the application’s text window. | “Background Color” on page 564 |
| --roottype | Appearance of the root window. | “Window Color” on page 657 |
| --rootcolor | Color of the root window. | “Window Color: Custom Color” on page 657 |
| --compression | Whether the Adaptive Internet Protocol (AIP) protocol compresses commands for transmission. | “Command Compression” on page 578 |
| --execution | Whether the AIP protocol always executes commands in order, or optimizes commands for performance reasons. | “Command Execution” on page 579 |
| --interlaced | Enables interlaced image transmission. | “Interlaced Images” on page 608 |
| --accel | Enables graphics acceleration for the application’s display. | “Graphics Acceleration” on page 601 |
| --delayed | Enables delayed updates of the application’s display. | “Delayed Updates” on page 590 |
| --ldapusers | Assigns the application to the specified LDAP users. | “Assigned User Profiles Tab” on page 558 |
| --ldapgroups | Assigns the application to the specified LDAP groups. | “Assigned User Profiles Tab” on page 558 |
| --ldapsearch | Assigns the application to the users that match the LDAP search criteria. | “Assigned User Profiles Tab” on page 558 |
| --env | Environment variable settings needed to run the application. | “Environment Variables” on page 594 |
| --login | The login script used to start the application. | “Login Script” on page 618 |

| Option | Description | More Information |
|-----------------|---|--|
| --winmgr | The Window Manager to use for the application. | "Window Manager" on page 659 |
| --resumetimeout | Number of minutes the application is resumable for. | "Application Resumability: Timeout" on page 551 |
| --middlemouse | Timeout for emulating a middle mouse button click using a two-button mouse. | "Middle Mouse Timeout" on page 625 |
| --windowclose | Effect on application session of closing the main application window. | "Window Close Action" on page 654 |
| --euro | Keycode mapping required by the application to support the euro character. | "Euro Character" on page 595 |
| --dpi | Monitor resolution that SGD reports to X applications. | "Monitor Resolution" on page 626 |
| --keepopen | Keep open the connection used to start the application. | "Keep Launch Connection Open" on page 608 |
| --lockkeymap | Prevents applications from changing keyboard mappings. | "Keyboard Map: Locked" on page 611 |
| --share | Enables resource sharing for similar application sessions. | "Share Resources Between Similar Sessions" on page 642 |
| --ssharguments | Command-line arguments for the ssh client. | "Connection Method: SSH Arguments" on page 584 |
| --file | Batch file used to create multiple objects within the organizational hierarchy. | |

To batch-create multiple objects, use the `--file` option. Use the other options to create a single object.

Examples

The following example creates a new 3270 application object for the application 3270cat. The emulator connects to the 3270 host warsaw.indigo-insurance.com.

```
$ tarantella object new_3270app \
--name "o=applications/ou=Finance/cn=3270cat" \
--width 1000 --height 800 \
--app /3270cat \
--hostname warsaw.indigo-insurance.com
```

```
tarantella object new_5250app
```

Creates one or more 5250 application objects. See [“5250 Application Object” on page 528](#).

Syntax

```
tarantella object new_5250app {
    --name obj
    --width pixels
    --height pixels
    [ --description text ]
    [ --args args ]
    [ --method telnet|ssh ]
    [ --resumable never|session|always ]
    [ --endswhen lastclient|windowmanager|windowmanageralone|
nowindows|loginscript|loginscriptnowindows ]
    [ --maxinstances 0|instances ]
    [ --displayusing clientwm|independent|kiosk|localx ]
    [ --maximize true|false ]
    [ --scalable true|false ]
    [ --icon icon_name ]
    [ --hints hint... ]
    [ --hostname host ]
    [ --portnumber tcp ]
    [ --tnclose 0|1|2|3 ]
    [ --kt pc|sun4|sun5|hp ]
    [ --bl 0|1|2|3|4 ]
    [ --ma true|false ]
    [ --mb true|false ]
    [ --si true|false ]
    [ --fg color ]
    [ --bg color ]
    [ --roottype default|custom ]
    [ --rootcolor color ]
    [ --compression automatic|on|off ]
    [ --execution automatic|inorder|optimized ]
    [ --interlaced automatic|on|off ]
    [ --accel true|false ]
    [ --delayed true|false ]
    [ --ldapusers user_dn... ]
    [ --ldapgroups group_dn... ]
    [ --ldapsearch search_string... ]
    [ --env setting... ]
    [ --login script ]
    [ --winmgr command... ]
    [ --resumetimeout mins ]
    [ --middlemouse ms ]
    [ --windowclose notifyapp|killapp|suspendsession|endsession ]
    [ --euro unicode|iso8859-15 ]
    [ --dpi monitordpi ]
    [ --keepopen true|false ]
    [ --lockkeymap true|false ]
}
```

```
[ --share true|false ]
[ --ssharguments args ]
} | --file file
```

Description

SGD uses the third-party TeemTalk for Unix emulator for 5250 applications. See the TeemTalk for Unix User's Guide supplied with SGD for details.

The following table shows the available options for this command.

| Option | Description | More Information |
|----------------|--|--|
| --name | The common name of the object in the SGD datastore. | "Name" on page 628 |
| --width | The width of the application, in pixels. | "Window Size: Width" on page 664 |
| --height | The height of the application, in pixels. | "Window Size: Height" on page 661 |
| --description | A text description of the object. | "Comment" on page 579 |
| --args | The command-line arguments to use when starting the application. | "Arguments for Command" on page 555 |
| --method | The mechanism used by the SGD server to access the application server and start the application. | "Connection Method" on page 581 |
| --resumable | Resumability behavior for the application. | "Application Resumability" on page 549 |
| --endswhen | When the application session ends. | "Session Termination" on page 640 |
| --maxinstances | The maximum number of instances of the application a user can run simultaneously. | "Number of Sessions" on page 630 |
| --displayusing | How the application is displayed to the user. | "Window Type" on page 665 |
| --maximize | The initial size of the application. | "Window Size: Client's Maximum Size" on page 660 |
| --scalable | Scale the application to fit the window in which it is displayed. | "Window Size: Scale to Fit Window" on page 663 |
| --icon | Webtop icon for the application. | "Icon" on page 606 |
| --hints | String containing additional name-value data for the application. | "Hints" on page 602 |
| --hostname | The AS/400 host that runs the application. | "Server Address" on page 639 |
| --portnumber | The TCP port number used to connect to the AS/400 host. | "Server Port" on page 639 |

| Option | Description | More Information |
|---------------|---|---|
| --tnclose | Behavior when telnet connection to the AS/400 host is closed. | “Connection Closed Action” on page 580 |
| --kt | Layout to use for mapping the keyboard to the terminal being emulated. | “Keyboard Type” on page 612 |
| --bl | Number of “soft button” levels to display. | “Displayed Soft Buttons” on page 591 |
| --ma | Maximizes the emulator window. | “Window Size: Maximized” on page 662 |
| --mb | Enables the application’s menu bar. | “Menu Bar” on page 625 |
| --si | Enables the File and Settings menu items. | “‘File’ and ‘Settings’ Menus” on page 596 |
| --fg | Text color in the application’s text window. | “Foreground Color” on page 599 |
| --bg | Background color of the application’s text window. | “Background Color” on page 564 |
| --roottype | Appearance of the root window. | “Window Color” on page 657 |
| --rootcolor | Color of the root window. | “Window Color: Custom Color” on page 657 |
| --compression | Whether the AIP protocol compresses commands for transmission. | “Command Compression” on page 578 |
| --execution | Whether the AIP always executes commands in order, or optimizes commands for performance reasons. | “Command Execution” on page 579 |
| --interlaced | Enables interlaced image transmission. | “Interlaced Images” on page 608 |
| --accel | Enables graphics acceleration for the application’s display. | “Graphics Acceleration” on page 601 |
| --delayed | Enables delayed updates of the application’s display. | “Delayed Updates” on page 590 |
| --ldapusers | Assigns the application to the specified LDAP users. | “Assigned User Profiles Tab” on page 558 |
| --ldapgroups | Assigns the application to the specified LDAP groups. | “Assigned User Profiles Tab” on page 558 |
| --ldapsearch | Assigns the application to the users that match the LDAP search criteria. | “Assigned User Profiles Tab” on page 558 |
| --env | Environment variable settings needed to run the application. | “Environment Variables” on page 594 |
| --login | The login script used to start the application. | “Login Script” on page 618 |
| --winmgr | The Window Manager to use for the application. | “Window Manager” on page 659 |

| Option | Description | More Information |
|------------------------------|---|--|
| <code>--resumetimeout</code> | Number of minutes the application is resumable for. | “Application Resumability: Timeout” on page 551 |
| <code>--middlemouse</code> | Timeout for emulating a middle mouse button click using a two-button mouse. | “Middle Mouse Timeout” on page 625 |
| <code>--windowclose</code> | Effect on application session of closing the main application window. | “Window Close Action” on page 654 |
| <code>--euro</code> | Keycode mapping required by the application to support the euro character. | “Euro Character” on page 595 |
| <code>--dpi</code> | Monitor resolution that SGD reports to X applications. | “Monitor Resolution” on page 626 |
| <code>--keepopen</code> | Keep open the connection used to start the application. | “Keep Launch Connection Open” on page 608 |
| <code>--lockkeymap</code> | Prevents applications from changing keyboard mappings. | “Keyboard Map: Locked” on page 611 |
| <code>--share</code> | Enables resource sharing for similar application sessions. | “Share Resources Between Similar Sessions” on page 642 |
| <code>--ssharguments</code> | Command-line arguments for the ssh client. | “Connection Method: SSH Arguments” on page 584 |
| <code>--file</code> | Batch file used to create multiple objects within the organizational hierarchy. | |

To batch-create multiple objects, use the `--file` option. Use the other options to create a single object.

Examples

The following example creates a new 5250 application object for the application 5250cat. The emulator runs on the application server prague, and connects to the AS/400 host warsaw.indigo-insurance.com.

```
$ tarantella object new_5250app \
--name "o=applications/ou=Finance/cn=5250cat" \
--width 400 --height 300 \
--app /5250cat \
--appserv "o=appservers/cn=prague" \
--hostname warsaw.indigo-insurance.com
```

tarantella object new_charapp

Creates one or more character application objects. See [“Character Application Object” on page 531](#).

Syntax

```
tarantella object new_charapp {
  --name obj
  --emulator scocon|vt420|wyse60
  --termtype type
  --width pixels
  --height pixels
  [ --description text ]
  [ --app pathname ]
  [ --args args ]
  [ --appserv obj... ]
  [ --method telnet|ssh ]
  [ --resumable never|session|always ]
  [ --maxinstances 0|instances ]
  [ --displayusing independent|kiosk ]
  [ --maximize true|false ]
  [ --cols cols ]
  [ --lines lines ]
  [ --icon icon_name ]
  [ --hints hint...]
  [ --font courier|helvetica|timesroman ]
  [ --fontsize points ]
  [ --fixedfont true|false ]
  [ --autowrap true|false ]
  [ --cursor off|block|underline ]
  [ --statusline none|indicator|hostmessages|standard|extended ]
  [ --scrollstyle line|multiple|smooth ]
  [ --border normal|indented|raised ]
  [ --answermsg message ]
  [ --appkeymode true|false ]
  [ --keypad numeric|application ]
  [ --cursorkeys application|cursor ]
  [ --escape 7-bit|8-bit ]
  [ --codepage 437|850|852|860|863|865|8859-1|8859-2|
Multinational|Mazovia|CP852 ]
  [ --ldapusers user_dn... ]
  [ --ldapgroups group_dn... ]
  [ --ldapsearch search_string... ]
  [ --loadbal default|cpu|memory|sessions ]
  [ --compression automatic|on|off ]
  [ --env setting... ]
  [ --login script ]
  [ --keymap keymap ]
  [ --attributemap attrmap ]
  [ --colormap colormap ]
  [ --resumetimeout mins ]
```

```
[ --windowclose suspendsession|endsession ]
[ --ssharguments args ]
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description | More Information |
|----------------|--|--|
| --name | The common name of the object in the SGD datastore. | "Name" on page 628 |
| --emulator | The type of emulation required for the application. | "Emulation Type" on page 593 |
| --termtype | The terminal type required for the application. | "Terminal Type" on page 645 |
| --width | The width of the application, in pixels. | "Window Size: Width" on page 664 |
| --height | The height of the application, in pixels. | "Window Size: Height" on page 661 |
| --description | A text description of the object. | "Comment" on page 579 |
| --app | Full path name of the application. | "Application Command" on page 547 |
| --args | The command-line arguments to use when starting the application. | "Arguments for Command" on page 555 |
| --appserv | The application servers that can run the application. | "Hosting Application Servers Tab" on page 604 |
| --method | The mechanism used by the SGD server to access the application server and start the application. | "Connection Method" on page 581 |
| --resumable | Resumability behavior for the application. | "Application Resumability" on page 549 |
| --maxinstances | The maximum number of instances of the application a user can run simultaneously. | "Number of Sessions" on page 630 |
| --displayusing | How the application is displayed to the user. | "Window Type" on page 665 |
| --maximize | The initial size of the application. | "Window Size: Client's Maximum Size" on page 660 |
| --cols | The number of columns in the terminal window. | "Window Size: Columns" on page 661 |
| --lines | The number of lines in the terminal window. | "Window Size: Lines" on page 662 |
| --icon | Webtop icon for the application. | "Icon" on page 606 |

| Option | Description | More Information |
|---------------|--|---|
| --hints | String containing additional name-value data for the application. | “Hints” on page 602 |
| --font | Determines the font family used within the terminal window for the application | “Font Family” on page 597 |
| --fontsize | Defines the font size in the terminal window. | “Font Size” on page 598 |
| --fixedfont | Uses the font size specified by --fontsize for the terminal window. | “Font Size: Fixed Font Size” on page 598 |
| --autowrap | Determines the behavior when a user types characters extending beyond the right edge of the terminal window. | “Line Wrapping” on page 613 |
| --cursor | Cursor style used for the application. | “Cursor” on page 588 |
| --statusline | Specifies the type of status line. | “Status Line” on page 643 |
| --scrollstyle | The scroll behavior of the terminal window. | “Scroll Style” on page 637 |
| --border | The border style for the terminal window. | “Border Style” on page 566 |
| --answermsg | Defines the message to return when an inquiry is sent from the application server to the emulator. | “Answerback Message” on page 546 |
| --appkeymode | Determines whether the application can change the codes generated by keys on the keyboard. | “Keyboard Codes Modification” on page 609 |
| --keypad | Specifies the behavior of the cursor keys. | “Numpad Codes Modification” on page 631 |
| --cursorkeys | Specifies the behavior of the cursor keys. | “Cursor Key Codes Modification” on page 588 |
| --escape | Specifies how escape sequences are sent from the emulator to the application server. | “Escape Sequences” on page 595 |
| --codepage | The code page to use for the emulator. | “Code Page” on page 573 |
| --ldapusers | Assigns the application to the specified LDAP users. | “Assigned User Profiles Tab” on page 558 |
| --ldapgroups | Assigns the application to the specified LDAP groups. | “Assigned User Profiles Tab” on page 558 |
| --ldapsearch | Assigns the application to the users that match the LDAP search criteria. | “Assigned User Profiles Tab” on page 558 |
| --loadbal | Load balancing algorithm to use. | “Application Load Balancing” on page 548 |

| Option | Description | More Information |
|-----------------|---|---|
| --compression | Whether the AIP protocol compresses commands for transmission. | “Command Compression” on page 578 |
| --env | Environment variable settings needed to run the application. | “Environment Variables” on page 594 |
| --login | The login script used to start the application. | “Login Script” on page 618 |
| --keymap | Path name of a keyboard map file. | “Keyboard Map” on page 610 |
| --attributemap | The attribute map to use for the application. | “Attribute Map” on page 563 |
| --colormap | The color map to use for the application. | “Color Map” on page 576 |
| --resumetimeout | Number of minutes the application is resumable for. | “Application Resumability: Timeout” on page 551 |
| --windowclose | Effect on application session of closing the main application window. | “Window Close Action” on page 654 |
| --ssharguments | Command-line arguments for the ssh client. | “Connection Method: SSH Arguments” on page 584 |
| --file | Batch file used to create multiple objects within the organizational hierarchy. | |

To batch-create multiple objects, use the `--file` option. Use the other options to create a single object.

Examples

The following example creates a character application object for the application Pers-o-dat. The application can be run on the application servers prague and london. Application server load balancing decides which application server to use.

```
$ tarantella object new_charapp \
--name "o=applications/cn=Pers-o-dat" \
--emulator vt420 --termtype vt220 \
--width 400 --height 300 \
--app /bin/persodat \
--appserv "o=appservers/cn=prague" \
"o=appservers/ou=IT/cn=london"
```

tarantella object new_container

Creates one or more Active Directory container objects. See [“Directory \(Light\): Active Directory Container Object” on page 535](#).

Syntax

```
tarantella object new_container { --name obj } | --file file
```

Description

To batch-create multiple objects, use the `--file` option. Use the other options to create a single object.

Examples

The following example creates a new Active Directory container object with name `Users`, within the `indigo-insurance.com` domain components.

```
$ tarantella object new_container \  
--name "dc=com/dc=indigo-insurance/cn=Users"
```

The following example creates two Active Directory container objects using a batch script defined as a “here-document”. You can alternatively store the batch script in a file, and reference it using `--file filename`.

```
$ tarantella object new_container --file - <<EOF  
--name "dc=com/dc=indigo-insurance/cn=Users"  
--name "dc=com/dc=indigo-insurance/cn=Applications"  
EOF
```

tarantella object new_dc

Creates one or more domain component objects. See [“Directory \(Light\): Domain Component Object” on page 535](#).

Syntax

```
tarantella object new_dc { --name obj } | --file file
```

Description

To batch-create multiple objects, use the `--file` option. Use the other options to create a single object.

Examples

The following example creates a new domain component object with name `com`, at the top level of the organizational hierarchy.

```
$ tarantella object new_dc --name "dc=com"
```

The following example creates two domain component objects using a batch script defined as a “here-document”. You can alternatively store the batch script in a file, and reference it using `--file filename`.

```
$ tarantella object new_dc --file - <<EOF
--name "dc=com"
--name "dc=com/dc=indigo-insurance"
EOF
```

tarantella object new_doc

Creates one or more document objects. See [“Document Object” on page 536](#).

Syntax

```
tarantella object new_doc {  
    --name obj  
    --url url  
    [ --description text ]  
    [ --newbrowser true|false ]  
    [ --icon icon_name ]  
    [ --hints hint... ]  
    [ --ldapusers user_dn... ]  
    [ --ldapgroups group_dn... ]  
    [ --ldapsearch search_string... ]  
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description | More Information |
|---------------|---|---|
| --name | The name of the document object. | “Name” on page 628 |
| --url | Uniform Resource Locator (URL) displayed when document object link is clicked. | “URL” on page 649 |
| --description | A text description of the object. | “Comment” on page 579 |
| --newbrowser | Displays the document in a new browser window. | “Window Type: New Browser Window” on page 668 |
| --icon | Webtop icon for the application. | “Icon” on page 606 |
| --hints | String containing additional name-value data for the application. | “Hints” on page 602 |
| --ldapusers | Assigns the application to the specified LDAP users. | “Assigned User Profiles Tab” on page 558 |
| --ldapgroups | Assigns the application to the specified LDAP groups. | “Assigned User Profiles Tab” on page 558 |
| --ldapsearch | Assigns the application to the users that match the LDAP search criteria. | “Assigned User Profiles Tab” on page 558 |
| --file | A file containing a batch of commands to configure application server load balancing. | |

To batch-create multiple objects, use the `--file` option. Use the other options to create a single object.

Examples

The following example creates a new document object named PhoneList.

```
$ tarantella object new_doc \  
--name "o=applications/ou=Finance/ou=Administration/cn=Phone List" \  
--url http://newyork.indigo-insurance.com \  
--newbrowser false
```

The following example creates two document objects using a batch script defined as a “here-document”. You can alternatively store the batch script in a file, and reference it using `--file filename`.

```
$ tarantella object new_doc --file - <<EOF  
--name "o=applications/ou=Finance/ou=Administration/cn=Phone List"  
--url http://newyork.indigo-insurance.com  
--newbrowser false  
--name "o=applications/cn=Indigo Insurance web site"  
--url http://www.indigo-insurance.com  
--newbrowser true  
EOF
```

tarantella object new_dynamicapp

Creates one or more dynamic application objects. See [“Dynamic Application Object” on page 539](#).

Syntax

```
tarantella object new_dynamicapp {  
    --name obj  
    --mapping [type=application]  
    [ --description text ]  
    [ --icon icon ]  
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description | More Information |
|---------------|--|--|
| --name | The name of the dynamic application. | “Name” on page 628 |
| --mapping | A mapping between a type and the name of an application object. | “Mappings Tab” on page 621 |
| --description | A text description of the object. | “Comment” on page 579 |
| --icon | Webtop icon for the application. | “Icon” on page 606 |
| --file | A file containing a batch of commands to create dynamic application objects. | |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example creates a new dynamic application object named `desktopApp`, with a mapping for a `windows` type.

```
$ tarantella object new_dynamicapp \  
--name "o=applications/cn=desktopApp" \  
--mapping windows="o=applications/cn=windows_desktop"
```

tarantella object new_group

Creates one or more group objects. See [“Group Object” on page 536](#).

Syntax

```
tarantella object new_group {  
    --name obj  
    [ --description text ]  
    [ --member obj... ]  
    [ --ldapusers user_dn... ]  
    [ --ldapgroups group_dn... ]  
    [ --ldapsearch search_string... ]  
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description | More Information |
|---------------|---|--|
| --name | The name of the group object. | "Name" on page 628 |
| --description | A text description of the object. | "Comment" on page 579 |
| --member | Member of the group object. | "Members Tab" on page 622 |
| --ldapusers | Assigns the application to the specified LDAP users. | "Assigned User Profiles Tab" on page 558 |
| --ldapgroups | Assigns the application to the specified LDAP groups. | "Assigned User Profiles Tab" on page 558 |
| --ldapsearch | Assigns the application to the users that match the LDAP search criteria. | "Assigned User Profiles Tab" on page 558 |
| --file | A file containing a batch of commands to configure application server load balancing. | |

To batch-create multiple objects, use the `--file` option. Use the other options to create a single object.

Examples

The following example creates a new group object with common name `WinHosts`, belonging to the organization object `appservers`. The group's members are the application server objects for the application servers `rome`, `brussels`, and `berlin`.

```
$ tarantella object new_group \  
  --name "o=appservers/cn=WinHosts" \  
  --member "cn=rome" \  
  --member "cn=brussels" \  
  --member "cn=berlin"
```

```
--member "o=appservers/ou=Sales/cn=rome" \  
--member "o=appservers/cn=brussels" \  
--member "o=appservers/ou=Marketing/cn=berlin"
```

The following example creates three group objects using a batch script defined as a “here-document”. The groups have no members. You can use [tarantella object add_member](#) to add members later from the command line. You can alternatively store the batch script in a file, and reference it using `--file filename`.

```
$ tarantella object new_group --file - <<EOF  
--name "o=appservers/cn=WinHosts"  
--name "o=appservers/cn=UNIXHosts"  
--name "o=applications/cn=Applications"  
EOF
```

tarantella object new_host

Creates one or more application server or dynamic application server objects. See “Application Server Object” on page 530 and “Dynamic Application Server Object” on page 539.

Syntax

For an application server object, use the following syntax:

```
tarantella object new_host {  
    --name obj  
    [ --address address ]  
    [ --description text ]  
    [ --ntdomain dom ]  
    [ --available true|false ]  
    [ --auth trytta|nevertrytta|default ]  
    [ --location location ]  
    [ --hostlocale ll_tt ]  
    [ --maxcount count ]  
    [ --userassign spec ]  
} | --file file
```

For a dynamic application server object, use the following syntax:

```
tarantella object new_host {  
    --name obj  
    --dynamic  
    [ --description text ]
```

```
[ --auth trytta|nevertrytta|default ]
[ --vsbclass classname ]
[ --vsbparams params ]
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description | More Information |
|---------------|---|---|
| --name | The names of objects you want to add assigned applications links for. | “Name” on page 628 |
| --address | Network address of the application server. | “Address” on page 545 |
| --dynamic | Create a dynamic application server. | “Dynamic Application Server Object” on page 539 |
| --description | A text description of the object. | “Comment” on page 579 |
| --ntdomain | The Windows domain used for application server authentication. | “Domain Name” on page 592 |
| --available | Specifies whether applications can run on this application server. | “Application Start” on page 554 |
| --auth | Specifies the policy for authenticating users on the application server, <i>if no password is already cached</i> for that server. | “Password Cache Usage” on page 632 |
| --location | String describing the location of the application server. Used for load balancing. | “Load Balancing Groups” on page 614 |
| --hostlocale | Default language setting for the application server. | “Prompt Locale” on page 635 |
| --vsbclass | Fully-qualified class name for the virtual server broker (VSB). Dynamic application servers only. | “Virtual Server Broker Class” on page 653 |
| --vsbparams | Parameters passed to the VSB. Dynamic application servers only. | “Virtual Server Broker Parameters” on page 654 |
| --maxcount | Maximum number of application sessions that can be run concurrently on the application server | “Maximum Count” on page 622 |
| --userassign | Specifies the users that can run applications on the application server | “User Assignment” on page 650 |
| --file | A file containing a batch of commands to add assigned applications links. | |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example creates a new application server object with common name *paris*, belonging to the organizational unit object *Finance*, which must already exist.

```
$ tarantella object new_host \  
--name "o=appservers/ou=Finance/cn=paris" \  
--address paris.indigo-insurance.com \  
--auth default \  
--location Europe-north
```

The following example creates a new dynamic application server object *MyBroker* that uses the User-defined SGD broker.

```
$ tarantella object new_host --dynamic \  
--name "o=appservers/cn=MyBroker" \  
--vsbclass com.sun.sgd.vsbim.UserDefinedSGDBroker
```

The following example creates three application server objects using a batch script defined as a “here-document”. Alternatively, you can store the batch script in a file, and reference it using `--file filename`.

```
$ tarantella object new_host --file - <<EOF  
--name "o=appservers/ou=Finance/cn=paris"  
--address paris.indigo-insurance.com  
--name "o=appservers/cn=brussels"  
--address brussels.indigo-insurance.com  
--name "o=appservers/ou=IT/cn=london"  
--address london.indigo-insurance.com  
EOF
```

tarantella object new_org

Syntax

Creates one or more organization objects. See [“Directory: Organization Object” on page 533](#).

```
tarantella object new_org {  
    --name obj  
    [ --description text ]  
    [ --conntype type_spec... ]  
    [ --cdm drive_spec... ]  
    [ --userprintingconfig true|false ]  
    [ --mapprinters 2|1|0 ]  
    [ --pdfenabled 1|0 ]  
    [ --pdfviewerenabled 1|0 ]  
    [ --pdfdriver driver_name ]  
    [ --pdfisdefault 1|0 ]  
    [ --pdfviewerisdefault 1|0 ]  
    [ --links obj... ]  
    [ --editprofile 2|1|0 ]  
    [ --clipboard 2|1|0 ]  
    [ --serialport 2|1|0 ]  
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description | More Information |
|----------------------|--|---|
| --name | The name of the organization object in the SGD datastore. | “Name” on page 628 |
| --description | A text description of the object. | “Comment” on page 579 |
| --conntype | The connections that are allowed between the client device and the SGD server. | “Connections” on page 582 |
| --cdm | The drives on a Microsoft Windows client device that can be accessed from applications running on application servers. | “Client Drive Mapping” on page 567 |
| --userprintingconfig | Enables user-specific printing configuration. | “Client Printing: Override” on page 570 |

| Option | Description | More Information |
|----------------------|---|--|
| --mapprinters | The client printers users can print to when printing from Windows applications. | “Client Printing” on page 569 |
| --pdfenabled | Enables users to print using the SGD “Universal PDF Printer” printer when printing from Windows applications. | “Universal PDF Printer” on page 647 |
| --pdfviewerenabled | Enables users to print using the SGD “Universal PDF Viewer” printer when printing from Windows applications. | “Universal PDF Viewer” on page 648 |
| --pdfdriver | The printer driver to use for SGD Portable Document Format (PDF) printing when printing from Windows applications. | “Postscript Printer Driver” on page 633 |
| --pdfisdefault | Sets the SGD “Universal PDF Printer” printer as the client’s default printer when printing from Windows applications. | “Make Universal PDF Printer the Default” on page 619 |
| --pdfviewerisdefault | Sets the SGD “Universal PDF Viewer” printer as the client’s default printer when printing from Windows applications. | “Make Universal PDF Viewer the Default” on page 620 |
| --links | Defines assigned applications links. | “Assigned Applications Tab” on page 556 |
| --editprofile | Whether users can create and edit profiles for use with the SGD Client. | “Client Profile Editing” on page 572 |
| --clipboard | Whether users can use copy and paste in Windows or X application sessions. | “Copy and Paste” on page 585 |
| --serialport | Whether users can access the serial ports on a client device from a Windows application. | “Serial Port Mapping” on page 637 |
| --file | Batch file used to create multiple objects within the organizational hierarchy. | |

To batch-create multiple objects, use the `--file` option. Use the other options to create a single object.

Examples

The following example creates a new organization object with name Indigo Insurance. Connections for all users in the organization are secure (SSL-based) unless the OU or user profile objects are configured to give a different type of connection.

```
$ tarantella object new_org \  
--name "o=Indigo Insurance" \  
--conntype '*:SSL'
```

The following example creates two organization objects using a batch script defined as a “here-document”. You can alternatively store the batch script in a file, and reference it using `--file filename`.

```
$ tarantella object new_org --file - <<EOF  
--name "o=Indigo Insurance"  
--name "o=Indigo Insurance Services"  
EOF
```

tarantella object new_orgunit

Creates one or more organizational unit (OU) objects. See [“Directory: Organizational Unit Object” on page 534](#).

Syntax

```
tarantella object new_orgunit {
    --name obj
    [ --description text ]
    [ --inherit true|false ]
    [ --conntype type_spec... ]
    [ --cdm drive_spec... ]
    [ --userprintingconfig 1|0 ]
    [ --mapprinters 2|1|0 ]
    [ --pdfenabled 1|0 ]
    [ --pdfviewerenabled 1|0 ]
    [ --pdfdriver driver_name ]
    [ --pdfisdefault 1|0 ]
    [ --pdfviewerisdefault 1|0 ]
    [ --links obj... ]
    [ --editprofile 2|1|0 ]
    [ --clipboard 2|1|0 ]
    [ --serialport 2|1|0 ]
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description | More Information |
|----------------------|--|---|
| --name | The name of the organizational unit object in the SGD datastore. | “Name” on page 628 |
| --description | A text description of the object. | “Comment” on page 579 |
| --inherit | Whether the assigned applications for the object also includes the assigned applications for the object’s parent. | “Inherit Assigned Applications from Parent” on page 607 |
| --conntype | The connections that are allowed between the client device and the SGD server. | “Connections” on page 582 |
| --cdm | The drives on a Microsoft Windows client device that can be accessed from applications running on application servers. | “Client Drive Mapping” on page 567 |
| --userprintingconfig | Enables user-specific printing configuration. | “Client Printing: Override” on page 570 |
| --mapprinters | The client printers users can print to when printing from Windows applications. | “Client Printing” on page 569 |

| Option | Description | More Information |
|-----------------------------------|---|--|
| <code>--pdfenabled</code> | Enables users to print using the SGD “Universal PDF Printer” printer when printing from Windows applications. | “Universal PDF Printer” on page 647 |
| <code>--pdfviewerenabled</code> | Enables users to print using the SGD “Universal PDF Viewer” printer when printing from Windows applications. | “Universal PDF Viewer” on page 648 |
| <code>--pdfdriver</code> | The printer driver to use for SGD PDF printing when printing from Windows applications. | “Postscript Printer Driver” on page 633 |
| <code>--pdfisdefault</code> | Sets the SGD “Universal PDF Printer” printer as the client’s default printer when printing from Windows applications. | “Make Universal PDF Printer the Default” on page 619 |
| <code>--pdfviewerisdefault</code> | Sets the SGD “Universal PDF Viewer” printer as the client’s default printer when printing from Windows applications. | “Make Universal PDF Viewer the Default” on page 620 |
| <code>--links</code> | Defines the assigned applications for an object. | “Assigned Applications Tab” on page 556 |
| <code>--editprofile</code> | Whether users can create and edit profiles for use with the SGD Client. | “Client Profile Editing” on page 572 |
| <code>--clipboard</code> | Whether users can use copy and paste in Windows or X application sessions. | “Copy and Paste” on page 585 |
| <code>--serialport</code> | Whether users can access the serial ports on a client device from a Windows application. | “Serial Port Mapping” on page 637 |
| <code>--file</code> | Batch file used to create multiple objects within the organizational hierarchy. | |

To batch-create multiple objects, use the `--file` option. Use the other options to create a single object.

Examples

The following example creates a new OU object with the name `IT`, belonging to the organization object `Indigo Insurance`, which must already exist. This OU inherits assigned applications from its parent, the organization object. Connections for all users in the OU are secure (SSL-based) unless their user profile objects are configured to give a different type of connection.

```
$ tarantella object new_orgunit \
  --name "o=Indigo Insurance/ou=IT" \
  --inherit true --conntype '*:SSL'
```

The following example creates three OU objects using a batch script defined as a “here-document”. The OU Administration belongs to the OU Finance, just created. You can alternatively store the batch script in a file, and reference it using `-file filename`.

```
$ tarantella object new_orgunit --file - <<EOF
--name "o=Indigo Insurance/ou=IT"
--name "o=Indigo Insurance/ou=Finance"
--name "o=Indigo Insurance/ou=Finance/ou=Administration"
EOF
```

tarantella object new_person

Creates one or more user profile objects. See [“User Profile Object”](#) on page 537.

Syntax

```
tarantella object new_person {
    --name obj
    --surname surname
    [ --description text ]
    [ --user user ]
    [ --email name@domain ]
    [ --ntdomain dom ]
    [ --inherit true|false ]
    [ --shared true|false ]
    [ --enabled true|false ]
    [ --conntype type_spec... ]
    [ --cdm drive_spec... ]
    [ --keymap keymap ]
    [ --bandwidth limit ]
    [ --links obj... ]
    [ --userprintingconfig 1|0 ]
    [ --mapprinters 2|1|0 ]
    [ --pdfenabled 1|0 ]
    [ --pdfviewerenabled 1|0 ]
    [ --pdfdriver driver_name ]
    [ --pdfisdefault 1|0 ]
    [ --pdfviewerisdefault 1|0 ]
    [ --editprofile 2|1|0 ]
    [ --clipboard 2|1|0 ]
    [ --serialport 2|1|0 ]
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description | More Information |
|---------------|---|---|
| --name | The common name of the object in the SGD datastore. | “Name” on page 628 |
| --surname | The surname, or family name, for the user profile. | “Surname” on page 644 |
| --description | A text description of the object. | “Comment” on page 579 |
| --user | The user name for the user profile. This is typically their UNIX® system user name. | “Login Name” on page 617 |
| --email | The email address for the user profile. | “Email Address” on page 593 |

| Option | Description | More Information |
|----------------------|--|---|
| --ntdomain | The Windows domain used for application server authentication. | “Domain Name” on page 592 |
| --inherit | Whether the assigned applications for the object also includes the assigned applications for the object’s parent. | “Inherit Assigned Applications from Parent” on page 607 |
| --shared | Whether the user profile object is used by a single user, or can be shared by multiple users in the form of a “guest” account. | “Login: Multiple” on page 616 |
| --enabled | Whether someone can log in using this user profile object. | “Login” on page 615 |
| --conntype | Defines the connections that are allowed between the client device and the SGD server. | “Connections” on page 582 |
| --cdm | The drives on a Microsoft Windows client device that users can access from applications. | “Client Drive Mapping” on page 567 |
| --keymap | The path name of a keyboard map file. | “Keyboard Map” on page 610 |
| --bandwidth | The maximum bandwidth this person can use for applications. | “Bandwidth Limit” on page 565 |
| --links | Defines the assigned applications for an object. | “Assigned Applications Tab” on page 556 |
| --userprintingconfig | Enables user-specific printing configuration. | “Client Printing: Override” on page 570 |
| --mapprinters | The client printers users can print to when printing from Windows applications. | “Client Printing” on page 569 |
| --pdfenabled | Enables users to print using the SGD “Universal PDF Printer” printer when printing from Windows applications. | “Universal PDF Printer” on page 647 |
| --pdfviewerenabled | Enables users to print using the SGD “Universal PDF Viewer” printer when printing from Windows applications. | “Universal PDF Viewer” on page 648 |
| --pdfdriver | The printer driver to use for SGD PDF printing when printing from Windows applications. | “Postscript Printer Driver” on page 633 |
| --pdfisdefault | Sets the SGD “Universal PDF Printer” printer as the client’s default printer when printing from Windows applications. | “Make Universal PDF Printer the Default” on page 619 |
| --pdfviewerisdefault | Sets the SGD “Universal PDF Viewer” printer as the client’s default printer when printing from Windows applications. | “Make Universal PDF Viewer the Default” on page 620 |
| --editprofile | Whether users can create and edit profiles for use with the SGD Client. | “Client Profile Editing” on page 572 |

| Option | Description | More Information |
|---------------------------|--|---|
| <code>--clipboard</code> | Whether users can use copy and paste in X or Windows application sessions. | “Copy and Paste” on page 585 |
| <code>--serialport</code> | Whether users can access the serial ports on a client device from a Windows application. | “Serial Port Mapping” on page 637 |
| <code>--file</code> | Batch file used to create multiple objects within the organizational hierarchy. | |

To batch-create multiple objects, use the `--file` option. Use the other options to create a single object.

Examples

The following example creates a new user profile object for Indigo Jones. Indigo inherits assigned applications from the organization object.

```
$ tarantella object new_person \
--name "o=Indigo Insurance/cn=Indigo Jones" \
--surname Jones --user indigo \
--email indigo@indigo-insurance.com --inherit true
```

The following example creates three user profile objects using a batch script defined as a “here-document”. You can alternatively store the batch script in a file, and reference it using `--file filename`.

```
$ tarantella object new_person --file - <<EOF
--name "o=Indigo Insurance/cn=Indigo Jones" --surname Jones
--name "o=Indigo Insurance/ou=IT/cn=Bill Orange" --surname Orange
--name "o=Indigo Insurance/ou=Finance/cn=Mulan Rouge" --surname Rouge
EOF
```

tarantella object new_windowsapp

Creates one or more Windows application objects. See [“Windows Application Object” on page 540](#).

Syntax

```
tarantella object new_windowsapp {
    --name obj
    --width pixels
    --height pixels
    [ --description text ]
    [ --winproto wts|none ]
    [ --trylocal true|false ]
    [ --ntdomain dom ]
    [ --app pathname ]
    [ --args args ]
    [ --appserv obj... ]
    [ --workingdir dir ]
    [ --resumable never|session|always ]
    [ --displayusing independent|kiosk|seamless ]
    [ --maxinstances 0|instances ]
    [ --maximize true|false ]
    [ --scalable true|false ]
    [ --depth 8|16|24|32 ]
    [ --icon icon_name ]
    [ --hints hint... ]
    [ --compression automatic|on|off ]
    [ --execution automatic|inorder|optimized ]
    [ --interlaced automatic|on|off ]
    [ --accel true|false ]
    [ --delayed true|false ]
    [ --winmgr command... ]
    [ --login script ]
    [ --protoargs args ]
    [ --resumetimeout mins ]
    [ --middlemouse ms ]
    [ --euro unicode|iso8859-15 ]
    [ --dpi monitordpi ]
    [ --lockkeymap true|false ]
    [ --loadbal default|cpu|memory|sessions ]
    [ --ldapusers user_dn... ]
    [ --ldapgroups group_dn... ]
    [ --ldapsearch search_string... ]
    [ --clipboardlevel level ]
    [ --windowclose suspendsession|endsession ]
    [ --cdm drive_spec... ]
    [ --appprintingconfig 1|0 ]
    [ --mapprinters 2|1|0 ]
    [ --pdfenabled 1|0 ]
    [ --pdfviewerenabled 1|0 ]
    [ --pdfdriver driver_name ]
    [ --pdfisdefault 1|0 ]
}
```

```
[ --pdfviewerisdefault 1|0 ]
[ --allowkioskescape true|false ]
[ --swmopts true|false ]
[ --console 1|0 ]
[ --remotewindowkeys true|false ]
[ --disablewallpaper 1|0 ]
[ --disablefullwindowdrag 1|0 ]
[ --disablemenuanimations 1|0 ]
[ --disabletheming 1|0 ]
[ --disablecursorshadow 1|0 ]
[ --disablecursorsettings 1|0 ]
[ --enablefontsmoothing 1|0 ]
[ --noprintprefs 1|0 ]
[ --remoteaudio 1|0 ]
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description | More Information |
|---------------|--|---|
| --name | The common name of the object in the SGD datastore. | “Name” on page 628 |
| --width | The width of the application, in pixels. | “Window Size: Width” on page 664 |
| --height | The height of the application, in pixels. | “Window Size: Height” on page 661 |
| --description | A text description of the object. | “Comment” on page 579 |
| --winproto | Whether to run the Windows application from an application server using Remote Desktop Protocol. | “SGD Remote Desktop Client” on page 642 |
| --trylocal | Try starting the application on the user’s client device. | “Local Client Launch” on page 615 |
| --ntdomain | The Windows domain to use for the application server authentication process. | “Domain Name” on page 592 |
| --app | Full path name of the application. | “Application Command” on page 547 |
| --args | The command-line arguments to use when starting the application. | “Arguments for Command” on page 555 |
| --appserv | The application servers that can run the application. | “Hosting Application Servers Tab” on page 604 |

| Option | Description | More Information |
|-----------------|--|--|
| --workingdir | Working directory to be used by the application. | “Working Directory” on page 668 |
| --resumable | Resumability behavior for the application. | “Application Resumability” on page 549 |
| --displayusing | How the application is displayed to the user. | “Window Type” on page 665 |
| --maxinstances | The maximum number of instances of the application a user can run simultaneously. | “Number of Sessions” on page 630 |
| --maximize | The initial size of the application. | “Window Size: Client’s Maximum Size” on page 660 |
| --scalable | Scale the application to fit the window in which it is displayed. | “Window Size: Scale to Fit Window” on page 663 |
| --depth | Color depth for the application. | “Color Depth” on page 574 |
| --icon | Webtop icon for the application. | “Icon” on page 606 |
| --hints | String containing additional name-value data for the application. | “Hints” on page 602 |
| --compression | Whether the AIP protocol compresses commands for transmission. | “Command Compression” on page 578 |
| --execution | Whether the AIP protocol always executes commands in order, or optimizes commands for performance reasons. | “Command Execution” on page 579 |
| --interlaced | Enables interlaced image transmission. | “Interlaced Images” on page 608 |
| --accel | Enables graphics acceleration for the application’s display. | “Graphics Acceleration” on page 601 |
| --delayed | Enables delayed updates of the application’s display. | “Delayed Updates” on page 590 |
| --winmgr | The Window Manager to use for the application. | “Window Manager” on page 659 |
| --login | The login script used to start the application. | “Login Script” on page 618 |
| --protoargs | Command-line arguments used for the SGD Remote Desktop Client. | “Arguments” on page 554 |
| --resumetimeout | Number of minutes the application is resumable for. | “Application Resumability: Timeout” on page 551 |
| --middlemouse | Timeout for emulating a middle mouse button click using a two-button mouse. | “Middle Mouse Timeout” on page 625 |

| Option | Description | More Information |
|---------------------|---|--|
| --euro | Keycode mapping required by the application to support the euro character. | “Euro Character” on page 595 |
| --dpi | Monitor resolution that SGD reports to X applications. | “Monitor Resolution” on page 626 |
| --lockkeymap | Prevents applications from changing keyboard mappings. | “Keyboard Map: Locked” on page 611 |
| --loadbal | Load balancing algorithm to use. | “Application Load Balancing” on page 548 |
| --ldapusers | Assigns the application to the specified LDAP users. | “Assigned User Profiles Tab” on page 558 |
| --ldapgroups | Assigns the application to the specified LDAP groups. | “Assigned User Profiles Tab” on page 558 |
| --ldapsearch | Assigns the application to the users that match the LDAP search criteria. | “Assigned User Profiles Tab” on page 558 |
| --clipboardlevel | Clipboard security level for the application. | “Copy and Paste: Application’s Clipboard Security Level” on page 587 |
| --windowclose | Effect on application session of closing the main application window. | “Window Close Action” on page 654 |
| --cdm | The drives on a Microsoft Windows client device that can be accessed from the application running on an application server. | “Client Drive Mapping” on page 567 |
| --appprintingconfig | Enables application-specific printing configuration. | “Client Printing: Override” on page 570 |
| --mapprinters | The client printers users can print to when printing from the application. | “Client Printing” on page 569 |
| --pdfenabled | Enables users to print using the SGD “Universal PDF Printer” printer when printing from the application. | “Universal PDF Printer” on page 647 |
| --pdfviewerenabled | Enables users to print using the SGD “Universal PDF Viewer” printer when printing from the application. | “Universal PDF Viewer” on page 648 |
| --pdfdriver | The printer driver to use for SGD PDF printing when printing from the application. | “Postscript Printer Driver” on page 633 |
| --pdfisdefault | Sets the SGD “Universal PDF Printer” printer as the client’s default printer when printing from the application. | “Make Universal PDF Printer the Default” on page 619 |

| Option | Description | More Information |
|--------------------------------------|--|---|
| <code>--pdfviewerisdefault</code> | Sets the SGD “Universal PDF Viewer” printer as the client’s default printer when printing from the application. | “Make Universal PDF Viewer the Default” on page 620 |
| <code>--allowkioskescape</code> | Enables a pull-down header for kiosk mode applications. | “Kiosk Mode Escape” on page 612 |
| <code>--swmopts</code> | Enables local window hierarchy for applications displayed in seamless windows mode. Needed for compatibility with some Borland applications. | “SWM Local Window Hierarchy” on page 645 |
| <code>--console</code> | Starts the application in console mode, also called remote administration mode. | “Console Mode” on page 585 |
| <code>--remotewindowkeys</code> | Sends window management key strokes to the remote session. | “Window Management Keys” on page 658 |
| <code>--disablewallpaper</code> | Disables background wallpaper. This can improve performance. | “Desktop Wallpaper” on page 591 |
| <code>--disablefullwindowdrag</code> | Disables the option to show the contents of a window when it is moved. This can improve performance. | “Full Window Drag” on page 600 |
| <code>--disablemenuanimations</code> | Disables transition effects for menus and tooltips. This can improve performance. | “Menu Animations” on page 624 |
| <code>--disabletheming</code> | Disables themes for the application. This can improve performance. | “Theming” on page 646 |
| <code>--disablecursorshadow</code> | Disables the mouse pointer shadow. This can improve performance. | “Cursor Shadow” on page 589 |
| <code>--disablecursorsettings</code> | Disables mouse pointer schemes and customizations. This can improve performance. | “Cursor Settings” on page 589 |
| <code>--enablefontsmoothing</code> | Enables font smoothing for text in the application. This can improve text readability, but can affect performance. | “Font Smoothing” on page 599 |
| <code>--noprintprefs</code> | Disables caching of printer preferences, such as paper size and page orientation. | “Printer Preference Caching” on page 635 |
| <code>--remoteaudio</code> | Leaves audio at the remote application server. | “Remote Audio” on page 636 |
| <code>--file</code> | Batch file used to create multiple objects within the organizational hierarchy. | |

To batch-create multiple objects, use the `--file` option. Use the other options to create a single object.

Examples

The following example creates a new Windows application object for the application Write-o-Win. The application runs on the application server rome.

```
$ tarantella object new_windowsapp \  
--name "o=applications/cn=Write-o-Win" \  
--width 1000 --height 800 \  
--app c:\\programs\\apps\\write.exe \  
--appserv "o=appservers/ou=Sales/cn=rome"
```

tarantella object new_xapp

Creates one or more X application objects. See [“X Application Object”](#) on page 542.

Syntax

```
tarantella object new_xapp {
    --name obj
    --width pixels
    --height pixels
    [ --description text ]
    [ --app pathname ]
    [ --args args ]
    [ --appserv obj... ]
    [ --method rexec|telnet|ssh ]
    [ --resumable never|session|always ]
    [ --endswhen lastclient|windowmanager|windowmanageralone|
nowindows|loginscript|loginscriptnowindows ]
    [ --maxinstances 0|instances ]
    [ --displayusing clientwm|independent|kiosk|localx ]
    [ --maximize true|false ]
    [ --scalable true|false ]
    [ --depth 8|16|24|16/8|24/8|8/16|8/24 ]
    [ --icon icon_name ]
    [ --hints hint... ]
    [ --clipboardlevel level ]
    [ --roottype default|custom ]
    [ --rootcolor color ]
    [ --compression automatic|on|off ]
    [ --execution automatic|inorder|optimized ]
    [ --quality automatic|best|24|21|18|16|15|12|9|6 ]
    [ --interlaced automatic|on|off ]
    [ --accel true|false ]
    [ --delayed true|false ]
    [ --ldapusers user_dn... ]
    [ --ldapgroups group_dn... ]
    [ --ldapsearch search_string... ]
    [ --loadbal default|cpu|memory|sessions ]
    [ --env setting... ]
    [ --login script ]
    [ --winmgr command... ]
    [ --resumetimeout mins ]
    [ --middlemouse ms ]
    [ --force3button true|false ]
    [ --windowclose notifyapp|killapp|suspendsession|endsession ]
    [ --euro unicode|iso8859-15 ]
    [ --dpi monitordpi ]
    [ --keepopen true|false ]
    [ --lockkeymap true|false ]
    [ --share true|false ]
    [ --securityextension true|false ]
    [ --ssharguments args ]
}
```

```
[ --unixaudiopreload true|false ]
[ --remotewindowkeys true|false ]
[ --allowkioskescape true|false ]
[ --allowsshdowngrade true|false ]
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description | More Information |
|----------------|--|--|
| --name | The common name of the object in the SGD datastore. | "Name" on page 628 |
| --width | The width of the application, in pixels. | "Window Size: Width" on page 664 |
| --height | The height of the application, in pixels. | "Window Size: Height" on page 661 |
| --description | A text description of the object. | "Comment" on page 579 |
| --app | Full path name of the application. | "Application Command" on page 547 |
| --args | The command-line arguments to use when starting the application. | "Arguments for Command" on page 555 |
| --appserv | The application servers that can run the application. | "Hosting Application Servers Tab" on page 604 |
| --method | The mechanism used by the SGD server to access the application server and start the application. | "Connection Method" on page 581 |
| --resumable | Resumability behavior for the application. | "Application Resumability" on page 549 |
| --endswhen | When the application session ends. | "Session Termination" on page 640 |
| --maxinstances | The maximum number of instances of the application a user can run simultaneously. | "Number of Sessions" on page 630 |
| --displayusing | How the application is displayed to the user. | "Window Type" on page 665 |
| --maximize | The initial size of the application. | "Window Size: Client's Maximum Size" on page 660 |
| --scalable | Scale the application to fit the window in which it is displayed. | "Window Size: Scale to Fit Window" on page 663 |
| --depth | Color depth for the application. | "Color Depth" on page 574 |
| --icon | Webtop icon for the application. | "Icon" on page 606 |
| --hints | String containing additional name-value data for the application. | "Hints" on page 602 |

| Option | Description | More Information |
|------------------|--|--|
| --clipboardlevel | Clipboard security level for the application. | “Copy and Paste: Application’s Clipboard Security Level” on page 587 |
| --roottype | Appearance of the root window. | “Window Color” on page 657 |
| --rootcolor | Color of the root window. | “Window Color: Custom Color” on page 657 |
| --compression | Whether the AIP protocol compresses commands for transmission. | “Command Compression” on page 578 |
| --execution | Whether the AIP protocol always executes commands in order, or optimizes commands for performance reasons. | “Command Execution” on page 579 |
| --quality | The effective color depth displayed on client devices. | “Color Quality” on page 576 |
| --interlaced | Enables interlaced image transmission. | “Interlaced Images” on page 608 |
| --accel | Enables graphics acceleration for the application’s display. | “Graphics Acceleration” on page 601 |
| --delayed | Enables delayed updates of the application’s display. | “Delayed Updates” on page 590 |
| --ldapusers | Assigns the application to the specified LDAP users. | “Assigned User Profiles Tab” on page 558 |
| --ldapgroups | Assigns the application to the specified LDAP groups. | “Assigned User Profiles Tab” on page 558 |
| --ldapsearch | Assigns the application to the users that match the LDAP search criteria. | “Assigned User Profiles Tab” on page 558 |
| --loadbal | Load balancing algorithm to use. | “Application Load Balancing” on page 548 |
| --env | Environment variable settings needed to run the application. | “Environment Variables” on page 594 |
| --login | The login script used to start the application. | “Login Script” on page 618 |
| --winmgr | The Window Manager to use for the application. | “Window Manager” on page 659 |
| --resumetimeout | Number of minutes the application is resumable for. | “Application Resumability: Timeout” on page 551 |
| --middlemouse | Timeout for emulating a middle mouse button click using a two-button mouse. | “Middle Mouse Timeout” on page 625 |
| --force3button | Specifies that the application only supports a 3-button mouse. | “Mouse” on page 627 |
| --windowclose | Effect on application session of closing the main application window. | “Window Close Action” on page 654 |

| Option | Description | More Information |
|----------------------------------|--|--|
| <code>--euro</code> | Keycode mapping required by the application to support the euro character. | “Euro Character” on page 595 |
| <code>--dpi</code> | Monitor resolution that SGD reports to X applications. | “Monitor Resolution” on page 626 |
| <code>--keepopen</code> | Keep open the connection used to start the application. | “Keep Launch Connection Open” on page 608 |
| <code>--lockkeymap</code> | Prevents applications from changing keyboard mappings. | “Keyboard Map: Locked” on page 611 |
| <code>--share</code> | Enables resource sharing for similar application sessions. | “Share Resources Between Similar Sessions” on page 642 |
| <code>--securityextension</code> | Enables the X Security Extension for the application. | “X Security Extension” on page 669 |
| <code>--ssharguments</code> | Command-line arguments for the ssh client. | “Connection Method: SSH Arguments” on page 584 |
| <code>--unixaudiopreload</code> | Enables the SGD audio redirection library. | “Audio Redirection Library” on page 563 |
| <code>--remotewindowkeys</code> | Sends window management key strokes to the remote session. | “Window Management Keys” on page 658 |
| <code>--allowkioskescape</code> | Enables a pull-down header for kiosk mode applications. | “Kiosk Mode Escape” on page 612 |
| <code>--allowsshdowngrade</code> | Enables SGD to try an X11 connection instead of SSH, when X11 forwarding is not configured or working. | “Allow SSH Downgrade” on page 546 |
| <code>--file</code> | Batch file used to create multiple objects within the organizational hierarchy. | |

To batch-create multiple objects, use the `--file` option. Use the other options to create a single object.

Examples

The following example creates a new X application object for the application XFinance. The application can be run on the application servers paris, bonn, or lisbon. Application server load balancing decides which one to use.

```
$ tarantella object new_xapp \
  --name "o=applications/ou=Finance/cn=XFinance" \
  --width 1000 --height 800 \
```

```
--app /usr/local/bin/xfinance \  
--appserv "o=appservers/ou=Finance/cn=paris" \  
"o=appservers/ou=Finance/cn=bonn" "o=appservers/cn=lisbon"
```

tarantella object remove_host

Removes application servers from the list of those that can run an application, for application server load balancing.

Syntax

```
tarantella object remove_host { --name obj...  
                                --host hobj...  
                                } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------|--|
| --name | Specifies the names of application objects you want to configure load balancing for. |
| --host | Specifies the names of application server objects you want to remove from the load balancing pool. |
| --file | Specifies a file containing a batch of commands to configure application server load balancing. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example removes the application server rome from the load balancing pool for the application Slide-o-Win.

```
$ tarantella object remove_host \  
--name "o=applications/cn=Slide-o-Win" \  
--host "o=appservers/ou=Sales/cn=rome"
```

The following example removes the group WinHosts from the load balancing pool for the applications Write-o-Win and Slide-o-Win. Load balancing is no longer performed across all the application servers in WinHosts.

```
$ tarantella object remove_host \  
--name "o=applications/cn=Write-o-Win" \  
"o=applications/cn=Slide-o-Win" \  
--host "o=appservers/cn=WinHosts"
```

tarantella object remove_link

Removes assigned applications links for an object.

Syntax

```
tarantella object remove_link { --name obj...  
                                --link lobj...  
                                } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------|--|
| --name | Specifies the names of objects you want to remove links for. |
| --link | Specifies the names of objects you want to remove links for. |
| --file | Specifies a file containing a batch of commands to remove links for. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example removes the Write-o-Win application from the assigned applications for Violet Carson.

```
$ tarantella object remove_link \  
--name "o=Indigo Insurance/ou=Sales/cn=Violet Carson" \  
--link "o=applications/cn=Write-o-Win"
```

The following example removes the group Applications from the assigned applications of the organizational units Sales and Marketing. Everyone who inherits assigned applications from one of these OUs no longer sees all the applications in their assigned applications. For example, if they belong to that OU and [Inherit Assigned Applications from Parent](#) is selected for their user profile object. However, they might still see an application if it is inherited from elsewhere.

```
$ tarantella object remove_link \  
--name "o=Indigo Insurance/ou=Sales" \  
"o=Indigo Insurance/ou=Marketing" \  
--link "o=applications/cn=Applications"
```

tarantella object remove_mapping

Removes type-application mappings for a dynamic application object. See [“Dynamic Application Object” on page 539](#).

Syntax

```
tarantella object remove_mapping {  
    --name obj  
    --mappingtype [type]  
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description | More Information |
|---------------|---|--|
| --name | The name of the dynamic application object. | "Name" on page 628 |
| --mappingtype | Type of mapping to be removed. | "Mappings Tab" on page 621 |
| --file | A file containing a batch of commands to remove mappings. | |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example removes a mapping for the windows type from a dynamic application object with common name winApp.

```
$ tarantella object remove_mapping \  
--name "o=applications/cn=winApp" \  
--mappingtype windows
```

The following example removes a mapping for the solaris type from a dynamic application object named desktopApp.

```
$ tarantella object remove_mapping \  
--name "o=applications/cn=desktopApp" \  
--mappingtype solaris
```

tarantella object remove_member

Removes objects from groups.

Syntax

```
tarantella object remove_member { --name obj...
                                   --member mobj...
                                   } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------|--|
| --name | Specifies the names of group objects you want to remove members from. |
| --member | Specifies the names of objects you want to remove from the groups. |
| --file | Specifies a file containing a batch of commands to remove group members. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example removes the Write-o-Win application from the group Applications.

```
$ tarantella object remove_member \
  --name "o=applications/cn=Applications" \
  --member "o=applications/cn=Write-o-Win"
```

The following example removes the three application server objects rome, brussels, and berlin from the group WinHosts.

```
$ tarantella object remove_member \
  --name "o=appservers/cn=WinHosts" \
```

```
--member "o=appservers/ou=Sales/cn=rome" \  
"o=appservers/cn=brussels" \  
"o=appservers/ou=Marketing/cn=berlin"
```

tarantella object rename

Renames or moves an object in the organizational hierarchy.

Syntax

```
tarantella object rename { --name obj...  
                           --newname newobj...  
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|-----------|--|
| --name | Specifies the name of the object you want to rename or move. |
| --newname | Specifies the new name of the object. |
| --file | Specifies a file containing a batch of commands to rename or move objects. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example renames the user profile object for Elizabeth Blue to Liz Blue.

```
$ tarantella object rename \  
  --name "o=Indigo Insurance/ou=Sales/cn=Elizabeth Blue" \  
  --newname "o=Indigo Insurance/ou=Sales/cn=Liz Blue"
```


The following example moves Ginger Butcher between the organizational units IT and Sales.

```
$ tarantella object rename \  
--name "o=Indigo Insurance/ou=IT/cn=Ginger Butcher" \  
--newname "o=Indigo Insurance/ou=Sales/cn=Ginger Butcher"
```

tarantella object script

Runs a batch script of `tarantella object` commands, or enables commands to be run interactively.

Syntax

```
tarantella object script
```

Description

The batch script consists of standard `tarantella object` commands, one per line, *without* the `tarantella object` prefix. For example, use `edit` rather than `tarantella object edit`.

The batch script can use a backslash (\) to break commands across multiple lines. Lines beginning with a hash (#) are treated as comments and ignored.

If you need to include quotes (") or a backslash (\) character in any of the values for the commands, you must backslash protect them. For example, to use "c:\ Program Files" as a value for the `--args` option, type the following:

```
--args "\"c:\\Program Files\""
```

The command reads from standard input. For example, you can use a “here-document” to run a batch script:

```
$ tarantella object script <<EOF  
commands  
EOF
```

If standard input is empty, you can run `tarantella object` commands interactively.

Examples

The following example adds the group Applications to the organizational units Sales and Marketing, and sets the Sales OU’s [Inherit Assigned Applications from Parent](#) attribute to false.

```
$ tarantella object script <<EOF
add_link
--name "o=Indigo Insurance/ou=Sales"
"o=Indigo Insurance/ou=Marketing"
--link "o=Indigo Insurance/cn=Applications"
edit
--name "o=Indigo Insurance/ou=Sales" --inherit false
EOF
```

The tarantella passcache Command

This command manipulates the application server password cache. SGD Administrators can create, modify, delete, and examine entries.

Syntax

```
tarantella passcache new | edit | list | delete
```

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|------------|--|---|
| new | Creates entries in the password cache. | “tarantella passcache new” on page 769 |
| edit | Modifies existing entries in the password cache. | “tarantella passcache edit” on page 765 |
| list | Lists the contents of the password cache. | “tarantella passcache list” on page 767 |
| delete | Deletes entries from the password cache. | “tarantella passcache delete” on page 763 |

Note – All commands include a `--help` option. You can use `tarantella passcache command --help` to get help on a specific command.

Examples

The following example creates a password cache entry for the SGD user Indigo Jones, on the application server represented by the application server object prague.

```
$ tarantella passcache new \  
--person "o=Indigo Insurance/cn=Indigo Jones" \  
--resource "o=appservers/cn=prague" \  
--resuser indigo --respass rainbow
```

The following example lists entries in the password cache for the SGD user Indigo Jones.

```
$ tarantella passcache list \  
--person "o=Indigo Insurance/cn=Indigo Jones"
```

tarantella passcache delete

Deletes entries in the application server password cache.

Note – You can also use this command to delete the decision to always use a smart card to authenticate to an application server.

Syntax

```
tarantella passcache delete { [--person pobj | --anon | --ldap ]
                             [ --resource resource ]
                             } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|------------|---|
| --person | Specifies the name of the user profile object to delete the password cache entry for. |
| --anon | Removes the password cache entry for all anonymous users. |
| --ldap | Deletes the password cache entry for a service object. |
| --resource | <p>Specifies the name of the application server, Microsoft Windows domain, or service object that the password cache entry applies to.</p> <p>The name can be one of the following:</p> <ul style="list-style-type: none">• An application server object, for example "o=appservers/cn=paris".• A DNS name, for example ".../_dns/paris.indigo-insurance.com".• A Windows domain, for example ".../_wns/indigo.dom".• ".../_array" to mean the array. This is used when caching the password used to log in to SGD. See “Password Cache Usage” on page 632.• A service object name. See “Using Service Objects” on page 90. |
| --file | Specifies a file containing password cache entries to delete. |

If neither --person, --anon, nor --ldap is specified, all password cache entries for the specified resource are deleted.

If --resource is not specified, all the password cache entries for the person, or anonymous user, are deleted.

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example deletes all password cache entries for the user Indigo Jones.

```
$ tarantella passcache delete \  
--person "o=Indigo Insurance/cn=Indigo Jones"
```

The following example deletes all password cache entries for anonymous users on the application server prague.indigo-insurance.com.

```
$ tarantella passcache delete \  
--anon --resource ../_dns/prague.indigo-insurance.com
```

The following example deletes the password cache entry for the east service object.

```
$ tarantella passcache delete \  
--ldap --resource east
```

tarantella passcache edit

Edits entries in the application server password cache.

Syntax

```
tarantella passcache edit {  
    { --person pobj | --anon | --ldap }  
    --resource resource  
    --resuser resuser  
    [ --respass respass ]  
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------|---|
| --person | Specifies the name of the user profile object to edit the password cache entry for. |
| --anon | Edits a password cache entry for anonymous users. |
| --ldap | Edits the password cache entry for a service object. |

| Option | Description |
|------------|--|
| --resource | Specifies the name of the application server, Microsoft Windows domain, or service object that the password cache entry applies to. The name can be one of the following: <ul style="list-style-type: none"> • An application server object, for example "o=appservers/cn=paris". • A DNS name, for example ".../_dns/paris.indigo-insurance.com". • A Windows domain, for example ".../_wns/indigo.dom". • ".../_array" to mean the array. This is used when caching the password used to log in to SGD. See "Password Cache Usage" on page 632. • A service object name. See "Using Service Objects" on page 90. |
| --resuser | Identifies the user name for the resource. |
| --respass | Specifies the password associated with --resuser. If you omit this option, you are prompted for the password. |
| --file | Specifies a file containing password cache entries to edit. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example edits the password cache entry for the SGD user Indigo Jones, on the application server represented by the application server object prague.

```
$ tarantella passcache edit \  
--person "o=Indigo Insurance/cn=Indigo Jones" \  
--resource "o=appservers/cn=prague" \  
--resuser indigo --respass rainbow
```

The following example edits the password cache entry for anonymous users on the application server paris.indigo-insurance.com.

```
$ tarantella passcache edit \  
--anon --resource .../_dns/paris.indigo-insurance.com
```

The following example creates a password cache entry for the mainldap service object, which is an LDAP service object.

```
$ tarantella passcache edit \  
--ldap --resource mainldap \  
--resuser cn=sgd-user,cn=Users,dc=example,dc=com \  
--respass rainbow
```

The following example edits the password cache entry for the east service object, which is an Active Directory service object.

```
$ tarantella passcache edit \  
--ldap --resource east \  
--resuser admin@east.example.com --respass rainbow
```

tarantella passcache list

Lists entries in the application server password cache.

Syntax

```
tarantella passcache list { [ --person pobj | --anon | --ldap ]  
                           [ --resource resource ]  
                           [ --resuser resuser ]  
                           [ --format text | xml ]  
                           } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------|---|
| --person | Specifies the name of the user profile object to list the password cache entry for. |
| --anon | Lists password cache entries for anonymous users. |
| --ldap | List the password cache entry for a service object. |

| Option | Description |
|-------------------------|---|
| <code>--resource</code> | Specifies the name of the application server, Microsoft Windows domain, or service object to list. The name can be one of the following: <ul style="list-style-type: none"> • An application server object, for example "o=appservers/cn=paris". • A DNS name, for example ".../_dns/paris.indigo-insurance.com". • A Windows domain, for example ".../_wns/indigo.dom". • ".../_array" to mean the array. This is used when caching the password used to log in to SGD. See “Password Cache Usage” on page 632. • A service object name. See “Using Service Objects” on page 90. |
| <code>--resuser</code> | Lists password cache entries for a particular user name. |
| <code>--format</code> | Specifies the output format. The default setting is <code>text</code> . |
| <code>--file</code> | Specifies a file containing password cache entries to list. |

If you omit all arguments, or just specify `--format`, all entries in the password cache are displayed.

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example lists entries in the password cache for the SGD user Indigo Jones.

```
$ tarantella passcache list \  
--person "o=Indigo Insurance/cn=Indigo Jones"
```

The following example lists all entries in the password cache.

```
$ tarantella passcache list
```

The following example lists the password cache entry for the `east` service object.

```
$ tarantella passcache list \  
--ldap --resource east
```


tarantella passcache new

Syntax

```
tarantella passcache new {
    { --person pobj | --anon | --ldap }
    --resource resource
    --resuser resuser
    [ --respass respass ]
} | --file file
```

Description

Adds entries to the application server password cache.

The following table shows available options for this command.

| Option | Description |
|------------|---|
| --person | Specifies the name of the user profile object to create a password cache entry for. |
| --anon | Creates a password cache entry for anonymous users. |
| --ldap | Creates a password cache entry for a service object. |
| --resource | <p>Specifies the name of the application server, Microsoft Windows domain, or service object that the password cache entry applies to.</p> <p>The name can be one of the following:</p> <ul style="list-style-type: none">• An application server object, for example "o=appservers/cn=paris".• A DNS name, for example ".../_dns/paris.indigo-insurance.com".• A Windows domain, for example ".../_wns/indigo.dom".• ".../_array" to mean the array. This is used when caching the password used to log in to SGD. See “Password Cache Usage” on page 632.• A service object name. See “Using Service Objects” on page 90. |
| --resuser | Identifies the user name for the resource. |
| --respass | <p>Specifies the password associated with --resuser.</p> <p>If you omit this option, you are prompted for the password.</p> |
| --file | Specifies a file containing entries to add to the password cache. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example creates a password cache entry for the SGD user Indigo Jones, on the application server represented by the application server object prague.

```
$ tarantella passcache new \  
--person "o=Indigo Insurance/cn=Indigo Jones" \  
--resource "o=appservers/cn=prague" \  
--resuser indigo --respass rainbow
```

The following example creates a password cache entry for anonymous users on the application server paris.indigo-insurance.com, prompting for the password.

```
$ tarantella passcache new --anon --resuser \  
--resource ../_dns/paris.indigo-insurance.com
```

The following example creates a password cache entry for the mainldap service object which is an LDAP service object.

```
$ tarantella passcache new \  
--ldap --resource mainldap \  
--resuser cn=sgd-user,cn=Users,dc=example,dc=com \  
--respass rainbow
```

The following example creates a password cache entry for the east service object which is an Active Directory service object.

```
$ tarantella passcache new \  
--ldap --resource east \  
--resuser admin@example.com --respass rainbow
```

The tarantella print Command

This command enables you to administer SGD printing services across the array.

Syntax

| | | | | | | |
|------------------------|------|--------|-------|--------|------|--|
| tarantella print start | stop | status | pause | resume | list | |
| cancel | move | | | | | |

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|------------|---|---|
| cancel | Cancels print jobs. | “tarantella print cancel” on page 772 |
| list | Lists print jobs. | “tarantella print list” on page 773 |
| move | Moves queued print jobs from one SGD server to another. | “tarantella print move” on page 774 |
| pause | Pauses printing temporarily. | “tarantella print pause” on page 775 |
| resume | Resumes printing. | “tarantella print resume” on page 776 |
| start | Starts printing services for the array. | “tarantella print start” on page 777 |
| status | Displays information about printing services. | “tarantella print status” on page 778 |
| stop | Stops printing services for the array. | “tarantella print stop” on page 779 |

Note – All commands include a `--help` option. You can use `tarantella print command --help` to get help on a specific command.

Examples

The following example starts SGD printing services for the array.

```
$ tarantella print start
```

The following example lists all print jobs for Bill Orange.

```
$ tarantella print list \  
--person "o=Indigo Insurance/ou=IT/cn=Bill Orange"
```

tarantella print cancel

Cancels SGD print jobs that are currently spooled.

You can run this command on any SGD server in the array.

Syntax

```
tarantella print cancel {  
    --all | --person pobj... [--server serv]  
    | --server serv }
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------|--|
| --all | Cancels all print jobs spooled across the array. |
| --jobid | Cancels jobs with the specified job IDs. |
| --person | Cancels jobs belonging to each specified user profile, which must be the name. If this is used without --server, SGD cancels all print jobs for each specified user profile. |
| --server | Cancels jobs on each SGD server listed. Use the peer DNS name for each server. If this is used with --person, SGD only cancels the print jobs for each specified user profile on each specified server. |

Examples

The following example cancels print jobs for Bill Orange.

```
$ tarantella print cancel \  
--person "o=Indigo Insurance/ou=IT/cn=Bill Orange"
```

The following example cancels all print jobs on the SGD server detroit.

```
$ tarantella print cancel --server "detroit.indigo-insurance.com"
```

tarantella print list

Lists print jobs currently spooled.

You can run this command on any SGD server in the array.

Syntax

```
tarantella print list
{ --jobid id... | [ --person pobj... ] [ --server serv... ] }
[ --format text|brief ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------|---|
| --jobid | Lists jobs with the specified job IDs. |
| --person | Lists jobs belonging to each specified person, which must be the name. |
| --server | Lists jobs for each specified SGD server. Use the peer DNS name for each server. If this is used with the --person option, SGD only lists the spooled print jobs for the specified user profile on that server. |
| --format | Specifies the output format. The "text" format displays a block of text for each print job, showing each print job attribute, for example the job ID and job owner, on a new line. A blank line separates each job. This is the default. The "brief" format shows print job attributes on one line. |

If you omit --jobid, and --person or --server are used, all print jobs across the array are listed.

Examples

The following example lists print jobs for Bill Orange, in "text" format.

```
$ tarantella print list \
--person "o=Indigo Insurance/ou=IT/cn=Bill Orange"
```

The following example lists print jobs in “text” format for Bill Orange and Rusty Spanner on the SGD servers detroit and chicago.

```
$ tarantella print list \  
--person "o=Indigo Insurance/ou=IT/cn=Bill Orange" \  
"o=Indigo Insurance/ou=IT/cn=Rusty Spanner" \  
--server "detroit.indigo-insurance.com" \  
"chicago.indigo-insurance.com"
```

tarantella print move

Moves queued print jobs from one SGD server to another.

If an SGD server is temporarily unavailable, you can use this command to move the print jobs that are “stranded” on that server.

Note – This command only moves the print jobs that are currently in the SGD print queue. The SGD print queue is located at `/opt/tarantella/var/print/queue`.

Syntax

```
tarantella print move --server serv [ --printer printer_name ]  
[ --cups { y | n | auto } ]  
[ --preserve ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|------------|--|
| --cups | Indicates that the SGD server you are moving print jobs from uses the Common UNIX Printing System (CUPS). If you do not use this option, a default of <i>auto</i> is assumed and this means SGD tries to detect whether CUPS is being used. If CUPS is incorrectly detected, use this option to tell SGD whether CUPS is being used (<i>y</i>) or not (<i>n</i>). |
| --preserve | Forces SGD to copy rather than move the print jobs to the target SGD server. The original print jobs are kept in the SGD print queue. Note - If SGD printing services are restarted on the original SGD server and the print jobs have not been deleted, they are printed. |
| --printer | The name of the printer on the SGD server where you are moving the print jobs. If you leave out this argument, a default of <i>tta_printer</i> is used. |
| --server | The fully qualified peer DNS name of the SGD server where you are moving the print jobs. |

Examples

The following example moves print jobs from the SGD server where the command is run to the printer called *tta_boston* on the SGD server *boston.indigo-insurance.com*.

```
$ tarantella print move \  
--server boston.indigo-insurance.com --printer tta_boston
```

tarantella print pause

You can run this command on any SGD server in the array.

Pauses SGD printing services. New print jobs continue to spool, but do not print until printing is resumed using `tarantella print resume`.

If `--server` is not used, this command pauses printing services across the array.

Note – Pausing printing services on individual SGD servers in the array can cause problems for users. Whenever you pause printing services, do so for the whole array.

Syntax

```
tarantella print pause [ --server serv... ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|-----------------------|--|
| <code>--server</code> | Pauses printing services on each SGD server listed. Use the peer DNS name for each server. |

Examples

The following example pauses printing services across the array.

```
$ tarantella print pause
```

The following example pauses printing services on the SGD servers `detroit` and `chicago`.

```
$ tarantella print pause \  
--server "detroit.indigo-insurance.com" \  
"chicago.indigo-insurance.com"
```

tarantella print resume

Resumes SGD printing services, previously suspended with `tarantella print pause`. Any spooled jobs begin to print.

If `--server` is not used, this command resumes printing services across the array.

You can run this command on any SGD server in the array.

Note – Resuming printing services on individual SGD servers in the array can cause problems for users. Whenever you resume printing services, do so for the whole array.

Syntax

```
tarantella print resume [ --server serv... ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|-----------------------|---|
| <code>--server</code> | Resumes printing services on each SGD server listed. Use the peer DNS name for each server. |

Examples

The following example resumes printing services across the array.

```
$ tarantella print resume
```

The following example resumes printing services on the SGD servers detroit and chicago.

```
$ tarantella print resume \  
--server "detroit.indigo-insurance.com" \  
"chicago.indigo-insurance.com"
```

tarantella print start

Starts SGD printing services. If `--server` is not used, this command starts printing services across the array.

You can run this command on any SGD server in the array.

Note – Starting printing services on individual SGD servers in the array can cause problems for users. Whenever you start printing services, do so for the whole array.

Syntax

```
tarantella print start [ --server serv... ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|-----------------------|--|
| <code>--server</code> | Starts printing services on each SGD server listed. Use the peer DNS name for each server. |

Examples

The following example starts printing services across the array.

```
$ tarantella print start
```

The following example starts printing services on the SGD server detroit.

```
$ tarantella print start --server "detroit.indigo-insurance.com"
```

tarantella print status

Displays information about SGD printing services, including the following:

- Whether printing services are available, not available, or paused.
- The number of print jobs spooled.

You can run this command on any SGD server in the array.

Syntax

```
tarantella print status [ --summary |  
                        --server serv |  
                        --namemapping ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|---------------|---|
| --summary | Shows information for the array. |
| --server | Shows information for the SGD server listed. Use the peer DNS name for the server. |
| --namemapping | <p>Lists all the current name mappings used for printing. The print name mapping table ensures that users can print from an application and then exit the application, without losing the print job.</p> <p>These name mappings expire in time. You can set the expiry timeout on the Global Settings → Security tab in the Administration Console.</p> |

Examples

The following example displays information about SGD printing services for the array.

```
$ tarantella print status --summary
```

tarantella print stop

Stops SGD printing services. Print jobs are not accepted and do not spool.

If --server is not used, this command stops printing services across the array.

You can run this command on any SGD server in the array.

Note – Stopping printing services on individual SGD servers in the array can cause problems for users. Whenever you stop printing services, do so for the whole array.

Syntax

```
tarantella print stop [ --server serv... ][ --purge ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------|--|
| --purge | Removes all pending print jobs. If you omit this, print jobs that are currently spooled are printed. |
| --server | Stops printing services on each SGD server listed. Use the peer DNS name for each server. |

Examples

The following example stops printing services across the array, removing all pending print jobs.

```
$ tarantella print stop --purge
```

The following example stops printing services on the SGD server detroit.

```
$ tarantella print stop --server "detroit.indigo-insurance.com"
```

The tarantella query Command

Examines the SGD server's log files.

Syntax

```
tarantella query audit | billing | errlog | uptime
```

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|------------|---|--|
| audit | Displays log entries matching some criteria. | “tarantella query audit” on page 781 |
| billing | Queries billing log files. | “tarantella query billing” on page 784 |
| errlog | Displays the error log of SGD components. | “tarantella query errlog” on page 786 |
| uptime | Displays how long an SGD server has been available for. | “tarantella query uptime” on page 787 |

Note – All commands include a `--help` option. You can use `tarantella query command --help` to get help on a specific command.

Examples

The following example displays all error logs.

```
# tarantella query errlog
```

The following example displays how long the SGD server `newyork.indigo-insurance.com` has been available.

```
# tarantella query uptime --server newyork.indigo-insurance.com
```

tarantella query audit

Displays all log entries matching some criteria.

Syntax

```
tarantella query audit
{ --app app | --person person | --host host | --filter filter }
[ --server arrayhost ]
[ --format text|csv|xml ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------|---|
| --app | Displays log entries referring to a specific application. Use the object name for the application. |
| --person | Displays log entries referring to a specific person. Use the object name for the person. |
| --host | Displays log entries referring to a specific SGD server. Use the object name or a peer DNS name for the server. |
| --filter | An RFC2254-compliant LDAP search filter to find matching entries to display. Enclose the filter in quotes. You can use the "=", "~=", "<=" and ">=" matching rules in the filter. |
| --server | Only show log entries from the specified SGD server. Use a peer DNS name. If you omit this option, log entries across the entire array are displayed. |
| --format | Specifies the output format. The default setting is <code>text</code> . If you select the text format, SGD formats the log output so that it is easy to read on screen, but it does not show every detail logged. Using the <code>csv</code> format shows every detail logged but it is only suitable for outputting to a file. |

Note – The output that you see depends on the Log Filter settings for the array. To produce log entries for processing by this command, make sure the Log Filter attribute on the Global Settings → Monitoring tab in the Administration Console includes at least one filter that outputs to a `.json` file.

Using a Filter

The attributes you use in the filter are the log fields used in the `.jsl` log files. The following table lists the commonly used attributes.

| Field Name | Description |
|-------------------|---|
| log-category | The logging component/sub-component/severity setting used in the log filters. For example, to find entries for a server/printing/* log filter, you can use a "(log-category=*printing*)" filter |
| log-date | The system date and time when the event took place. The format is yyyy/MM/dd HH:mm:ss.SSS. |
| log-ip-address | The IP address of a client or server associated with an event. |
| log-keyword | The keyword for auditable events. |
| log-localhost | The peer DNS name of the SGD server where the event took place. |
| log-pid | The process ID of the event. |
| log-security-type | The type of security used on a connection, std or ssl. |
| log-systime | The system Coordinated Universal Time (UTC) time, in milliseconds, when the event took place. |
| log-tfn-name | The name of an object associated with an event. For example, starting an application session can record the name of the user, the application and the SGD server. |

Note – A complete list of all the log fields is available in the `/opt/tarantella/var/serverresources/schema/log.at.conf` schema file.

Examples

The following example displays all log entries for the UNIX system user indigo that were logged on the SGD server boston.indigo-insurance.com.

```
# tarantella query audit \  
--person ../_user/indigo --server boston.indigo-insurance.com
```

The following example outputs all log entries that refer to the Write-o-Win application, in comma-separated values (CSV) format.

```
# tarantella query audit \  
--app "o=applications/cn=Write-o-win" --format csv
```

The following example outputs all log errors that occurred on or after 23 October 2003 for the Write-o-Win application, in human-readable text format.

```
# tarantella query audit \  
--filter "(&(log-category=*error*)(log-tfn-name=o=  
applications/cn=Write-o-win) (log-date>=2003/10/23 00:00:00.0))"\  
--format text
```

tarantella query billing

Outputs billing information for the array, or for a subset of the array, over a time period. Information is displayed on screen in CSV format.

Syntax

```
tarantella query billing { --full | --sessions | --summary }  
                        --start date  
                        --days days  
                        --end date  
                        [ --servers arrayhost... ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|------------|---|
| --full | Displays detailed information for all user sessions and application sessions. |
| --sessions | Displays information for all application sessions. |
| --summary | Displays a short summary of billing information and an application session summary. |
| --start | Specifies the start of the billing period. The format is YYYY/MM/DD, for example, "2000/05/01". |

| Option | Description |
|-----------|--|
| --days | Specifies the number of days from the date specified by --start to display billing information. |
| --end | Specifies the end of the billing period. The format is YYYY/MM/DD, for example, "2000/05/02". The end date is <i>exclusive</i> . This means, for example, that --start 2001/01/19 --end 2001/01/23 is the same as --start 2001/01/19 --days 4. Both examples query data covering the 19th, 20th, 21st, and 22nd. |
| --servers | Only reports billing information from the named SGD servers. Use peer DNS names. If you omit --servers, billing information across the array is reported. |

The billing files are written at midnight *local time* each day.

You must run this command on the primary server in the array.

Note – You must enable billing services, see “Billing Service” on page 482, and restart all SGD servers in the array before any data is logged.

Examples

The following example displays billing information for the entire array, for the 30 days from May 1, 2000.

```
# tarantella query billing --full \
  --start "2000/05/01" --days 30
```

The following example displays a short summary of billing information for the servers prague and paris, for the 30 days from January 1 2000.

```
# tarantella query billing --summary \
  --start "2000/01/01" --days 30 \
  -- servers prague.indigo-insurance.com paris.indigo-insurance.com
```

The following example displays billing information for all application sessions for the entire array for the period January 19 2001 to January 22 2001 and outputs the results to a file called Sessions.csv.

```
# tarantella query billing --sessions \
  --start "2000/01/19" --end "2000/01/23" > sessions.csv
```

tarantella query errlog

Displays the error logs of SGD components.

Syntax

```
tarantella query errlog
    [ all|xpe|tpe|print|jserver|pemanager|proxy|wm ]
    [ --server arrayhost ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|---|---|
| all xpe tpe print jserver pemanager proxy wm | Specifies the component error log to display. Use all, the default, to display all error logs. |
| --server | Displays error logs from the named SGD server. Use a peer DNS name. If you omit this option, error logs from all SGD servers in the array are displayed. |

Note – To display error log information from the JServer component, make sure the Log Filter attribute on the Global Settings → Monitoring tab of the Administration Console includes at least one filter that outputs to an `error.log` file. The attribute does include this, by default.

Examples

The following example displays all error logs.

```
$ tarantella query errlog
```

The following example displays the X Protocol Engine error log on the SGD server `newyork.indigo-insurance.com`.

```
$ tarantella query errlog xpe \  
--server newyork.indigo-insurance.com
```

tarantella query uptime

Displays how long SGD servers have been available for.

Syntax

```
tarantella query uptime [ --server array/host ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|-----------------------|--|
| <code>--server</code> | Display information for the specified SGD server. Use a peer DNS name. If you omit this option, information for all SGD servers in the array is displayed. |

Examples

The following example displays how long all SGD servers in the array have been available for.

```
$ tarantella query uptime
```

The tarantella restart Command

Stops and then restarts services on the SGD server, prompting if users are currently connected.

Syntax

```
tarantella restart [ --warm | --force | --kill ] [ --quiet ]  
                  [ --http | --https ] [ --servlet ]
```

```
tarantella restart sgd [ --warm | --force | --kill ] [ --quiet ]
```

```
tarantella restart webserver [ --http | --https ] [ --servlet ]
```

Description

If no subcommands are specified, this command restarts both the SGD server and the SGD web server.



Caution – Never use the UNIX platform `kill` command to stop SGD services.

The following table shows the available options for this command.

| Option | Description |
|------------------------|--|
| <code>--force</code> | Tries harder to stop SGD services. |
| <code>--kill</code> | Kills the process IDs used by SGD services. Only use this option if you are having difficulty stopping the SGD server by other means. |
| <code>--quiet</code> | Does not prompt. Stops SGD services even if users are connected. |
| <code>--warm</code> | Tries a “warm restart” of the SGD server. This restarts the JServer component, without affecting other components. This has no effect on user sessions or application sessions. Only use this option if no users can log in to SGD or launch applications and no specific reason is found. |
| <code>--http</code> | Restarts Hypertext Transport Protocol (HTTP) services (Apache). |
| <code>--https</code> | Restarts HTTP over SSL (HTTPS) services (Apache). Requires a valid SSL certificate for the SGD web server. |
| <code>--servlet</code> | Restarts Java™ Servlet extension and JavaServer Pages™ (JSP™) technology services (Tomcat). |

Stopping an SGD server causes all user sessions and application sessions, including suspended application sessions, to be terminated.

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|------------|--|--|
| sgd | Restarts <i>only</i> the SGD server. | “tarantella restart sgd” on page 789 |
| webserver | Restarts <i>only</i> the SGD web server. | “tarantella restart webserver” on page 790 |

Note – All commands include a `--help` option. You can use `tarantella restart subcommand --help` to get help on a specific command.

Examples

The following example restarts the SGD server *and* the SGD web server in HTTP mode. SGD does not display a confirmation message if users are currently connected.

```
# tarantella restart --quiet --http
```

```
tarantella restart sgd
```

Stops and restarts only the SGD server.

Syntax

```
tarantella restart sgd [ --warm | --force | --kill ] [ --quiet ]
```

Description

Stops and restarts the SGD server.

The following table shows the available options for this command.

| Option | Description |
|----------------------|--|
| <code>--force</code> | Tries harder to stop SGD services. |
| <code>--kill</code> | Kills the process IDs used by SGD services. Only use this option if you are having difficulty stopping the SGD server by other means. |
| <code>--quiet</code> | Does not prompt. Stops SGD services even if users are connected. |
| <code>--warm</code> | Tries a “warm restart” of the SGD server. This restarts the JServer component, without affecting other components. This has no effect on user sessions or application sessions. Only use this option if no users can log in to SGD or launch applications and no specific reason is found. |

Examples

The following example restarts the SGD server, without displaying a confirmation message if users are currently connected.

```
# tarantella restart sgd --quiet
```

tarantella restart webserver

Stops and restarts only the SGD web server.

Syntax

```
tarantella restart webserver [ --http | --https ] [ --servlet ]
```

Description

If you do not use any command options, the command restarts both the SGD web server (Apache), and Java Servlet extension and JSP technology services (Tomcat).

The following table shows the available options for this command.

| Option | Description |
|-----------|--|
| --http | Restarts HTTP services (Apache). |
| --https | Restarts HTTPS services (Apache). Requires a valid SSL certificate for the SGD web server. |
| --servlet | Restarts Java Servlet extension and JSP technology services (Tomcat). |

Note – If you restart both Apache and Tomcat using separate subsequent commands, you must restart Tomcat first.

Examples

The following example restarts Apache and Tomcat.

```
# tarantella restart webserver
```

The tarantella role Command

You use this command to give users specific roles, and to give them assigned applications that apply to that role.

Syntax

| | | | | | | | | | | | | | | | | |
|------------|------|----------|--|------------|--|------|--|------------|--|------|--|--------------|--|-------------|--|---------------|
| tarantella | role | add_link | | add_member | | list | | list_links | | list | | list_members | | remove_link | | remove_member |
|------------|------|----------|--|------------|--|------|--|------------|--|------|--|--------------|--|-------------|--|---------------|

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|---------------|---|---|
| add_link | Adds assigned applications links for occupants of particular roles. | "tarantella role add_link" on page 793 |
| add_member | Adds occupants to particular roles. | "tarantella role add_member" on page 794 |
| list | Lists and describes all available roles. | "tarantella role list" on page 795 |
| list_links | Lists the assigned applications links for occupants of particular roles. | "tarantella role list_links" on page 795 |
| list_members | Lists the occupants of particular roles. | "tarantella role list_members" on page 796 |
| remove_link | Removes assigned applications links for users occupying particular roles. | "tarantella role remove_link" on page 797 |
| remove_member | Removes occupants from particular roles. | "tarantella role remove_member" on page 798 |

Note – All commands include a --help option. You can use tarantella role subcommand --help to get help on a specific command.

Examples

The following example lists all available roles.

```
$ tarantella role list
```


The following example adds a link for the application Indigo Time to the assigned applications of users occupying the Global Administrators role.

```
$ tarantella role add_link \  
--role global \  
--link "o=applications/cn=Indigo Time"
```

tarantella role add_link

Adds assigned applications links for users occupying particular roles.

Syntax

```
tarantella role add_link { --role rolename  
                           --link lobj...  
                           } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------|---|
| --role | Specifies the name of a role, for example global. Use <code>tarantella role list</code> to find out the available roles. |
| --link | Specifies the names of objects to add to the assigned applications of users occupying the role, for example, <code>o=applications/cn=Indigo Time</code> . |
| --file | Specifies a file containing a batch of commands to add assigned applications links for users with a particular role. |

Note – Make sure you quote any object names containing spaces, for example, `"o=Indigo Insurance"`.

Examples

The following example adds a link for the application Indigo Time to the assigned applications of users occupying the Global Administrators role.

```
$ tarantella role add_link \  
--role global \  
--link "o=applications/cn=Indigo Time"
```

tarantella role add_member

Adds occupants to particular roles.

Syntax

```
tarantella role add_member { --role rolename  
                             --member mobj...  
                             } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------|--|
| --role | Specifies the name of a role, for example <code>global</code> . Use <code>tarantella role list</code> to find out the available roles. |
| --member | Specifies the names of user profile objects or profile objects for the users you want to occupy the role. |
| --file | Specifies a file containing a batch of commands to add occupants to particular roles. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example adds Sid Cerise to the Global Administrators role.

```
$ tarantella role add_member \  
--role global \  
--member "o=Indigo Insurance/ou=Finance/cn=Sid Cerise"
```

tarantella role list

Lists and describes all available roles, including the name of the role object applicable to each role.

Syntax

```
tarantella role list
```

Description

Use the short name, for example “global”, with other `tarantella role` commands.

Examples

The following example lists all available roles.

```
$ tarantella role list
```

tarantella role list_links

Lists the assigned applications links for occupants of particular roles. The name for each link is shown.

Syntax

```
tarantella role list_links --role rolename | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------|--|
| --role | Specifies the name of a role, for example <code>global</code> . Use <code>tarantella role list</code> to find out the available roles. |
| --file | Specifies a file containing a batch of commands to list the assigned applications for role occupants. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example lists the assigned applications for occupants of the Global Administrators role.

```
$ tarantella role list_links --role global
```

tarantella role list_members

Lists the occupants of particular roles. The name for each member is shown.

Syntax

```
tarantella role list_members --role rolename | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------|--|
| --role | Specifies the name of a role, for example <code>global</code> . Use <code>tarantella role list</code> to find out the available roles. |
| --file | Specifies a file containing a batch of commands to list the occupants of a particular role. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example lists the names of all occupants of the Global Administrators role.

```
$ tarantella role list_members --role global
```

tarantella role remove_link

Removes assigned applications links for users occupying particular roles.

Syntax

```
tarantella role remove_link { --role rolename
                             --link lobj...
                             } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------|---|
| --role | Specifies the name of a role, for example <code>global</code> . Use <code>tarantella role list</code> to find out the available roles. |
| --link | Specifies the names of assigned applications links to remove for users occupying the role. For example, <code>o=applications/cn=Indigo Time</code> . |
| --file | Specifies a file containing a batch of commands to remove assigned applications links of users with a particular role. |

Note – Make sure you quote any object names containing spaces, for example, `"o=Indigo Insurance"`.

Examples

The following example removes the Write-o-Win application from the assigned applications of members of the Global Administrators role.

```
$ tarantella role remove_link \  
--role global \  
--link "o=applications/cn=Write-o-Win"
```

tarantella role remove_member

Removes occupants from particular roles.

Syntax

```
tarantella role remove_member { --role rolename
                                --member mobj...
                                } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------|---|
| --role | Specifies the name of a role, for example global. Use tarantella role list to find out the available roles. |
| --member | Specifies the names of objects for the users you do not want to occupy the role. |
| --file | Specifies a file containing a batch of commands to remove occupants from a particular role. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example removes Sid Cerise from the Global Administrators role.

```
$ tarantella role remove_member \
--role global \
--member "o=Indigo Insurance/ou=Finance/cn=Sid Cerise"
```



The tarantella security Command

Controls SGD security services and manages server certificates.

Syntax

| | | | |
|------------------------------|-------------|---------|-------------|
| tarantella security certinfo | certrequest | certuse | customca |
| decryptkey | disable | enable | fingerprint |
| peerca | selfsign | start | stop |

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|-------------|---|---|
| certinfo | Displays information about an SSL certificate or Certificate Signing Request (CSR), and optionally checks whether a specified private key matches the public key contained in a particular SSL certificate. | “tarantella security certinfo” on page 801 |
| certrequest | Creates a CSR and a corresponding key pair, which you use to obtain an SSL certificate for use with SGD security services. | “tarantella security certrequest” on page 803 |
| certuse | Installs an SSL certificate, or specifies the location of an installed certificate, for use with SGD security services. | “tarantella security certuse” on page 805 |
| customca | Installs a root certificate for a custom CA for use with SGD security services. | “tarantella security customca” on page 807 |
| decryptkey | Decrypts an encrypted private key so that you can use it with SGD. | “tarantella security decryptkey” on page 808 |
| disable | If an SGD server has been secured using the <code>tarantella security enable</code> command, restores the security settings to their previous state. | “tarantella security disable” on page 809 |
| enable | Makes an SGD server secure. | “tarantella security enable” on page 810 |
| fingerprint | Displays the fingerprint of the CA certificate installed on the SGD server. | “tarantella security fingerprint” on page 812 |
| peerca | Shows, imports, or exports the primary server’s CA certificate used for secure intra-array communication. | “tarantella security peerca” on page 813 |

| Subcommand | Description | More Information |
|------------|--|--|
| selfsign | Generates and installs a self-signed server SSL certificate. | “tarantella security selfsign” on page 813 |
| start | Enables secure (SSL) connections. Users who require secure connections are given them. | “tarantella security start” on page 814 |
| stop | Disables secure (SSL) connections. Users configured for secure connections are given standard connections instead. | “tarantella security stop” on page 815 |

Note – All commands include a `--help` option. You can use `tarantella security subcommand --help` to get help on a specific command.

Examples

The following example displays information about a CSR in `/tmp/boston.csr`.

```
# tarantella security certinfo --csrfile /tmp/boston.csr
```

The following example decrypts the key `/opt/keys/key1`, which is stored in Definite Encoding Rules (DER) format, placing the decrypted key in `/opt/keys/key2`.

```
# tarantella security decryptkey \
  --enckey /opt/keys/key1 \
  --deckey /opt/keys/key2 \
  --format DER
```

tarantella security certinfo

Displays information about an installed SSL certificate (`--certfile`) or a Certificate Signing Request (`--csrfile`).

Syntax

```
tarantella security certinfo  
[ --certfile certfile [ --keyfile keyfile ] ]  
[ --full ]
```

```
tarantella security certinfo --csrfile csrfile [ --full ]
```

Description

This command can also check whether a specified private key matches the public key in a particular SSL certificate. In other words, the public key can decrypt text encrypted with the private key.

Use the first form of this command without specifying a *certfile* and *keyfile* to check the key and SSL certificate installed using the `tarantella security certuse` command.

The following table shows the available options for this command.

| Option | Description |
|------------|--|
| --certfile | <p>Specifies the location of a file containing a server SSL certificate. The command displays information about this certificate, including the following:</p> <ul style="list-style-type: none">• Information about the server and your organization.• Alternative DNS names for the server.• Credentials of the CA that validated the server SSL certificate.• Dates for which the SSL certificate is valid. <p>If you omit <code>--certfile</code>, the command displays information about the SSL certificate and key installed in the <code>/opt/tarantella/var/tsp</code> directory.</p> <p>You must specify the full path to the SSL certificate file.</p> |
| --keyfile | <p>Specifies the location of a private key. The command checks whether a private key matches the public key contained in the SSL certificate file.</p> <p>You must specify the full path to the key file.</p> |
| --csrfile | <p>Specifies the location of a file containing a CSR. The command displays information about this CSR, including the following:</p> <ul style="list-style-type: none">• The DNS name, or chosen common name, of the server the CSR is for.• Alternative DNS names for the server.• Your organization's name and location. <p>You must specify the full path to the CSR file.</p> |
| --full | <p>Displays more detailed information about the specified SSL certificate or CSR, for example, the contents of the public keys they contain.</p> |

Examples

The following example displays detailed information about the SSL certificate in the `/opt/certs/newyork.cert` file.

```
# tarantella security certinfo \  
--certfile /opt/certs/newyork.cert \  
--full
```

The following example displays information about the SSL certificate in `/opt/certs/boston.cert`, and checks that the private key `/opt/keys/boston.key` matches the public key contained in that SSL certificate.

```
# tarantella security certinfo \  
--certfile /opt/certs/boston.cert \  
--keyfile /opt/keys/boston.key
```

The following example displays information about the CSR in `/tmp/boston.csr`.

```
# tarantella security certinfo \  
--csrfile /tmp/boston.csr
```

tarantella security certrequest

Generates a CSR, and a public and private key pair.

Syntax

```
tarantella security certrequest --country country  
                                --state state  
                                --orgname org  
                                [ --ouname ou ]  
                                [ --email email ]  
                                [ --locality locality ]  
                                [ --keylength length ]
```

Description

You send the generated CSR to a supported CA to obtain an SSL certificate for use with SGD security services.

Note the following:

- If your CA lets you change the host name stored in the SSL certificate, make sure the SSL certificate contains a fully qualified DNS name. For example, `boston.indigo-insurance.com`, not `boston`.
- If the SGD server has multiple DNS names, for example, it is known by different names inside and outside a firewall, you can specify the additional DNS names as *subject alternative names* for the SSL certificate. This enables you to associate more than one DNS name with the SSL certificate.
- Make a copy of the private key and CSR generated by this command and keep them in a safe, secure location. Key information is stored in the `/opt/tarantella/var/tsp` directory. *If your private key is lost or damaged, you will be unable to use any SSL certificate you obtain using the CSR.*
- This command generates a new CSR and key pair each time you run it. If you generate a new CSR with this command, the previous CSR is overwritten and the new private key is stored in the file `/opt/tarantella/var/tsp/key.pending.pem`.

You can use the `tarantella security certinfo` command to display information about SSL certificates and CSRs.

If you do not specify `--ouname`, `--email` or `--locality` SGD omits that information from the CSR. There are no default values.

The options that can be used for this command are as follows.

| Option | Description |
|--------------------------|--|
| <code>--country</code> | Specifies the country where your organization is located. Use ISO 3166 country codes. For example, use <code>US</code> for the United States or <code>DE</code> for Germany. |
| <code>--state</code> | Specifies the state or province where your organization is located. Do not use abbreviations here. For example, use <code>Massachusetts</code> rather than <code>Mass.</code> or <code>MA</code> . |
| <code>--orgname</code> | Specifies the official, legal name of your organization. |
| <code>--ouname</code> | Specifies the name of a organizational unit (OU) within your organization, if required. If you do not need to specify an OU, you can use this setting to specify a less formal organization name. |
| <code>--email</code> | Specifies your business email address. This address is used for correspondence between you and the CA you send the CSR to. |
| <code>--locality</code> | Specifies the city or principality where your organization is located, if needed. |
| <code>--keylength</code> | Specifies the length of the key pair. The default is 1024. |

Note – Make sure you quote any object names containing spaces, for example, "o=Indigo Insurance".

Examples

The following example generates a CSR for Indigo Insurance, located in Massachusetts, with contact Bill Orange.

```
# tarantella security certrequest \  
--country US \  
--state MA \  
--orgname "Indigo Insurance" \  
--email "orange@indigo-insurance.com"
```

tarantella security certuse

Installs a server SSL certificate, or specifies the location of a previously installed SSL certificate, to be used by SGD security services.

Syntax

```
tarantella security certuse
```

```
tarantella security certuse --certfile cfile [ --keyfile kfile ]
```

Description

SSL certificates must be Base 64-encoded Privacy Enhanced Mail (PEM) format, with a header line including "BEGIN CERTIFICATE", as used by OpenSSL.

If no arguments are specified, this command reads the SSL certificate from standard input and installs it in `/opt/tarantella/var/tsp`.

After installing an SSL certificate, you must restart SGD using the `tarantella restart` command.

The following table shows the available options for this command.

| Option | Description |
|-------------------------|---|
| <code>--certfile</code> | <p>Specifies the location of a file containing the SSL certificate. If no <code>--keyfile</code> argument is specified, SGD assumes that the <code>tarantella security certrequest</code> command was used to generate the private key. You can use this option as follows:</p> <ul style="list-style-type: none">• To tell SGD about an SSL certificate you have already installed for use with another product, such as a web server. In this case, SGD <i>makes symbolic links to, not copies of</i>, the SSL certificate file and key file, if specified.• To install an SSL certificate received from a CA after generating a CSR using <code>tarantella security certrequest</code>. In this case, SGD installs the SSL certificate in <code>/opt/tarantella/var/tsp</code> for use with SGD security services. <p>You must specify the full path to the SSL certificate file.</p> |
| <code>--keyfile</code> | <p>Specifies the location of a file containing the private key for the SSL certificate specified by <code>--certfile</code>.</p> <p>Use this option to tell SGD about a private key you have already. If you used the <code>tarantella security certrequest</code> command to generate a CSR and obtain an SSL certificate, you do not need to use this option. You must specify the full path to the key file.</p> |

Examples

The following command installs an SSL certificate, which is saved in a temporary file `/tmp/cert`, and uses the private key generated when the `tarantella security certrequest` command was used to generate the CSR:

```
# tarantella security certuse < /tmp/cert
```

The following command installs an SSL certificate, which is stored in `/opt/certs/cert`, and a private key, which is stored in `/opt/keys/key`. The `tarantella security certrequest` command was *not* used to generate the CSR.

```
# tarantella security certuse \  
--certfile /opt/certs/cert \  
--keyfile /opt/keys/key
```

tarantella security customca

Installs or removes a root certificate for a custom CA for use with SGD security services.

Syntax

```
tarantella security customca
```

```
tarantella security customca --rootfile carootfile | --remove
```

Description

CA certificates must be Base 64-encoded PEM-format, with a header line including "BEGIN CERTIFICATE", as used by OpenSSL.

If no arguments are specified, this command reads the root certificate from standard input.

The following table shows the available options for this command.

| Option | Description |
|------------|--|
| --rootfile | Specifies the location of a file containing the CA's root certificate. Details are copied to /opt/tarantella/var/tsp for use by SGD security services. You must specify the full path to the root certificate file. |
| --remove | Removes any custom CA's root certificate currently installed for use with SGD security services. |

This command also imports the CA certificate into the CA certificate truststore for the SGD server. This is the /opt/tarantella/bin/jre/lib/security/cacerts file.

Examples

The following example installs a CA's root certificate from the file/tmp/rootcert, which you can then delete.

```
# tarantella security customca --rootfile /tmp/rootcert
```

tarantella security decryptkey

Decrypts an encrypted private key so that you can use it with SGD. This enables you to use an SSL certificate that you are already using with another product such as a web server, rather than obtaining a separate SSL certificate for use exclusively with SGD.

Syntax

```
tarantella security decryptkey --enckey enckeyfile
                                --deckey deckeyfile
                                [ --format PEM|DER ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------|---|
| --enckey | Specifies the location of the encrypted private key that you want to decrypt. Only keys encrypted by a product that uses SSLeay or OpenSSL certificate libraries can be decrypted. You must specify the full path to the encrypted private key file. |
| --deckey | Specifies a file where the decrypted key is stored. Note - For security reasons, it is very important to restrict access to private keys, especially when stored in an unencrypted form. Access to private keys by unauthorized users can result in a serious security breach. Store private keys accordingly. You must specify the full path to the decrypted key file. |
| --format | Specifies the format the encrypted key is stored in. Defaults to PEM. |

Note – You can only decrypt private keys that were originally encrypted by a product that uses SSLeay or OpenSSL certificate libraries.

See the [tarantella security certuse](#) command for information about how to share server SSL certificates in this way.

Examples

The following example decrypts the key `/opt/keys/key1`, which is stored in DER format, placing the decrypted key in `/opt/keys/key2`.

```
# tarantella security decryptkey \  
--enckey /opt/keys/key1 \  
--deckey /opt/keys/key2 \  
--format DER
```

tarantella security disable

If an SGD server has been secured using the `tarantella security enable` command, this command restores the security settings to their previous state.

Syntax

```
tarantella security disable
```

Description

Use this command to disable security services for an SGD server.

The following limitations apply for this command:

- **Automatic security configurations only.** Only use this command if you used the `tarantella security enable` command to enable security automatically on the SGD host. See [“tarantella security enable” on page 810](#) for more details.
- **Standalone servers only.** The SGD server must not be joined with other SGD servers in an array. If the SGD server is a member of an array, detach the SGD server from the array before using this command.

The command restores the security settings of an SGD server to their previous non-secure state. Any server SSL certificates or CA certificates are not removed.

Examples

The following example disables security services for an SGD server.

```
# tarantella security disable
```

tarantella security enable

Makes an SGD server secure.

Syntax

```
tarantella security enable
```

```
tarantella security enable --certfile cfile
                             [ --keyfile kfile ]
                             [ --rootfile carootfile ]
                             [ --firewalltraversal on|off ]
```

Description

Use this command to secure an SGD server.

The following limitations apply for this command:

- **New installations only.** The SGD installation must be a fresh installation and there has been no attempt to configure SGD security services.
- **Standalone servers only.** The SGD server must not be joined with other SGD servers in an array. If the SGD server is a member of an array, detach the SGD server from the array before using this command.

Use the `--certfile` option to specify a server SSL certificate to install. Certificates must be Base 64-encoded PEM-format, with a header line including "BEGIN CERTIFICATE", as used by OpenSSL.

If you omit the `--certfile` option, this command generates and installs a self-signed server SSL certificate. Only use self-signed server SSL certificates for test purposes.

If you use the `--certfile` option and the `--keyfile` option together, SGD creates *symbolic links* to the specified SSL certificate and key files.

Use the `--rootfile` option to install the CA certificate if the SSL certificate is signed by an unsupported CA. This option also imports the CA certificate into the CA certificate truststore for the SGD server. This is the `/opt/tarantella/bin/jre/lib/security/cacerts` file.

Use the `--firewalltraversal` option to enable or disable the SGD server for firewall traversal. SGD servers configured for firewall traversal cannot be used with the SGD Gateway.

If you have attempted to configure security previously, the `tarantella security enable` command has no effect. The command exits with an error message, indicating that security settings have been modified previously.

Ensure that the SGD server is running before you use this command. You can use the `tarantella status` command to show the current status of an SGD server.

The following table shows the available options for this command.

| Option | Description |
|----------------------------------|--|
| <code>--certfile</code> | Specifies the location of a file containing the SSL certificate. You must specify the full path to the SSL certificate file. |
| <code>--keyfile</code> | Specifies the location of a file containing the private key for the SSL certificate specified by <code>--certfile</code> . Use this option to tell SGD about a private key you have already. If you used the <code>tarantella security certrequest</code> command to generate a CSR and obtain an SSL certificate, you do not need to use this option. You must specify the full path to the key file. |
| <code>--rootfile</code> | Specifies the location of a file containing the CA's root certificate. Details are copied to <code>/opt/tarantella/var/tsp</code> for use by SGD security services. You must specify the full path to the CA root certificate file. |
| <code>--firewalltraversal</code> | Configures the SGD server for firewall traversal. If you do not specify this option, firewall traversal is enabled by default. |

If you use this command to secure an SGD server, the `tarantella security disable` command can be used to restore the security settings to their previous state.

Examples

The following example secures the SGD server, installs the specified SSL certificate, and uses the private key generated when the `tarantella security certrequest` command was used to generate a CSR:

```
# tarantella security enable \  
--certfile /opt/certs/cert
```

The following example secures the SGD server, and installs the specified SSL certificate and private key. A CA root certificate is also installed. The `tarantella security certrequest` command was *not* used to generate a CSR.

```
# tarantella security enable \  
--certfile /opt/certs/cert \  
--keyfile /opt/keys/key \  
--rootfile /tmp/rootcert
```

The following example secures the SGD server and installs a self-signed SSL certificate. Firewall traversal is not enabled for the SGD server.

```
# tarantella security enable \  
--firewalltraversal off
```

tarantella security fingerprint

Displays the fingerprint of the CA certificate installed on the SGD server.

Syntax

```
tarantella security fingerprint
```

Description

This command displays the fingerprint of the CA certificate installed using the `tarantella security customca` command.

If the SSL certificate for an SGD server is signed by a supported CA, you do not need to install a CA certificate.

If a server SSL certificate is not installed on the SGD server, this command shows the fingerprint of the built-in SGD CA certificate

Examples

The following example displays the fingerprint of the CA certificate installed on the SGD server.

```
# tarantella security fingerprint
```

tarantella security peerca

Shows, imports or exports the primary server's CA certificate used for secure intra-array communication.

Syntax

tarantella security peerca [--show | --import *hostname* | --export]

Description

The following table shows the available options for this command.

| Option | Description |
|----------|---|
| --show | Displays the primary server's CA certificate for the array. |
| --import | Import the CA certificate from the specified server. |
| --export | Export the CA certificate from this server. |

Examples

The following example shows the primary server's CA certificate for the array.

tarantella security peerca --show

tarantella security selfsign

Generates and installs a self-signed server SSL certificate.

Syntax

```
tarantella security selfsign
```

Description

Generates and installs a self-signed server SSL certificate. You must run the `tarantella security certrequest` command before using this command.

Only use self-signed server SSL certificates in a test environment because self-signed SSL certificates are not truly secure. While a self-signed server SSL certificate can be used to give users secure connections, users have no guarantee that the server they are connecting to is genuine. Self-signed certificates are valid for 365 days.

Examples

The following example generates and installs a self-signed server SSL certificate.

```
# tarantella security selfsign
```

tarantella security start

Starts security services on the SGD server where the command is run. Secure (SSL-based) connections are given to those users configured to require them.

Syntax

```
tarantella security start
```

Description

To enable secure connections to a particular SGD server you must already have installed an SSL certificate for that server.

Secure connections are enabled for the SGD server where the command is run.

Ensure that the SGD server is running before you use this command. You can use the `tarantella status` command to show the current status of an SGD server.

Examples

The following example enables secure connections for the SGD server where the command is run.

```
# tarantella security start
```

tarantella security stop

Disables security services on the SGD server where the command is run. Users configured to require secure (SSL-based) connections are given standard connections instead, if available.

Syntax

```
tarantella security stop [ --keep ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------|--|
| --keep | Specifies that any existing secure connections are preserved. If omitted, all secure connections are closed. |

If you run the command without any options, secure connections are disabled for the SGD server where the command is run.

Examples

The following example disables security services for the SGD server where the command is run, but preserves any existing secure connections.

```
# tarantella security stop --keep
```

The `tarantella service` Command

You use this command to manage service objects used for the following SGD authentication mechanisms:

- Active Directory authentication
- LDAP authentication
- Third-party authentication (if the LDAP search option is enabled)

See [“Using Service Objects” on page 90](#) for more details about service objects.

Syntax

| | | | |
|--|-------------------|-------------------|------------------|
| <code>tarantella service delete</code> | <code>edit</code> | <code>list</code> | <code>new</code> |
|--|-------------------|-------------------|------------------|

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|---------------------|--|---|
| <code>delete</code> | Deletes a service object | “tarantella service delete” on page 817 |
| <code>edit</code> | Edits a service object | “tarantella service edit” on page 817 |
| <code>list</code> | Lists service objects and their attributes | “tarantella service list” on page 821 |
| <code>new</code> | Creates a new service object | “tarantella service list” on page 821 |

Note – All commands include a `--help` option. You can use `tarantella service subcommand --help` to get help on a specific command.

Examples

The following example lists all of the available service objects and their attribute values.

| |
|---------------------------------------|
| <pre>\$ tarantella service list</pre> |
|---------------------------------------|

The following example deletes the mainldap service object.

```
$ tarantella service delete --name mainldap
```

tarantella service delete

Deletes a service object.

See [“Using Service Objects” on page 90](#) for more details about service objects.

Syntax

```
tarantella service delete { --name obj...  
                           } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|--------|--|
| --name | Specifies the name of the service object to delete. This can be a space-separated list of names. |
| --file | Specifies a file containing a batch of commands to delete service objects. |

Examples

The following example deletes the east service object.

```
$ tarantella service delete --name east
```

tarantella service edit

Edits one or more attributes for a service object.

See [“Using Service Objects” on page 90](#) for more details about service objects.

Syntax

```
tarantella service edit {
    --name obj
    [ --url url... ]
    [ --position pos ]
    [ --enabled 0|1 ]
    [ --operation-timeout timeout ]
    [ --base-domain domain ]
    [ --default-domain domain ]
    [ --black-list list ]
    [ --white-list list ]
    [ --security-mode ""|clientcerts]
    [ --auth-mode kerberos|ssl ]
    [ --site-aware 0|1 ]
    [ --site-name name ]
    [ --check-pwd-policy 0|1 ]
    [ --pwd-expiry-warn-threshold threshold ]
    [ --pwd-expiry-fail-threshold threshold ]
    [ --domain-list domains ]
    [ --password-update-mode ldapuser|ldapadmin]
    [ --lookupcache-timeout timeout ]
    [ --ad-alwaysusegc 0|1 ]
    [ --suffix-mappings mappings ]
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|------------|--|
| --name | The name of the service object to edit. See “Name” on page 447 for more details. |
| --url | The URLs of the LDAP directories or the URL of an Active Directory forest. The URL(s) must be unique. Different service objects cannot use the same URL(s). See “URLs” on page 448 for more details. |
| --position | A number that specifies the position of the service object in the list of service objects. The number 1 means first position in the list. |
| --enabled | Whether the service object is enabled for use for authentication. See “Enabled” on page 448 for more details. |

| Option | Description |
|----------------------------------|---|
| <code>--operation-timeout</code> | <p>Period of time, in seconds, to wait for a directory server to respond to an LDAP operation.</p> <p>See “LDAP Operation Timeout” on page 99 for more details.</p> |
| <code>--base-domain</code> | <p>The domain that SGD uses for Active Directory authentication if users only supply a partial domain when they log in.</p> <p>See “Active Directory Base Domain” on page 450 for more details.</p> <p>Applies only to Active Directory service objects.</p> |
| <code>--default-domain</code> | <p>The domain that SGD uses for Active Directory authentication if users do not supply a domain when they log in.</p> <p>See “Active Directory Default Domain” on page 450 for more details.</p> <p>Applies only to Active Directory service objects.</p> |
| <code>--black-list</code> | <p>A list of Active Directory servers which are never used for LDAP queries.</p> <p>See “Blacklists” on page 97 for more details.</p> <p>Applies only to Active Directory service objects.</p> |
| <code>--white-list</code> | <p>A list of Active Directory servers which are always used for LDAP queries. Servers not included in the list cannot be used.</p> <p>See “Whitelists” on page 97 for more details.</p> <p>Applies only to Active Directory service objects.</p> |
| <code>--security-mode</code> | <p>Whether client certificates are used to authenticate the SSL connection to an Active Directory server. This option is only used if <code>--auth-mode</code> is SSL.</p> <p>See “SSL Connections to Active Directory” on page 56 for more details.</p> <p>Applies only to Active Directory service objects.</p> |
| <code>--auth-mode</code> | <p>The mechanism used to secure the connection to an Active Directory server, either Kerberos or SSL. Kerberos is used by default.</p> <p>See “SSL Connections to Active Directory” on page 56 for more details.</p> <p>Applies only to Active Directory service objects.</p> |
| <code>--site-aware</code> | <p>Enables site awareness for the service object. If <code>--site-name</code> is not set, SGD attempts to discover site information automatically by contacting the global catalog.</p> <p>See “Sites” on page 96 for more details.</p> <p>Applies only to Active Directory service objects.</p> |
| <code>--site-name</code> | <p>A site name for the service object. This option is only used if <code>--site-aware</code> is enabled.</p> <p>See “Sites” on page 96 for more details.</p> <p>Applies only to Active Directory service objects.</p> |
| <code>--check-pwd-policy</code> | <p>Whether a user’s password policy should be checked at authentication time. This option is used to enable LDAP password expiry features.</p> <p>See “Password Expiry” on page 95 for more details.</p> |

| Option | Description |
|--|--|
| <code>--pwd-expiry-warn-threshold</code> | The period of time, in seconds, before password expiry where a warning message is shown on the webtop. See “Password Expiry” on page 95 for more details. |
| <code>--pwd-expiry-fail-threshold</code> | The period of time, in seconds, before password expiry where authentication is denied for a user and they are forced to update their password. See “Password Expiry” on page 95 for more details. |
| <code>--domain-list</code> | Defines a list of domains to be contacted when SGD starts. See “Domain Lists” on page 99 for more details. Applies only to Active Directory service objects. |
| <code>--password-update-mode</code> | Determines how aged passwords are handled. The default setting is <code>ldapuser</code> , meaning that passwords are updated using the authenticated user credentials. This results in a password <i>change</i> . A setting of <code>ldapadmin</code> means that passwords are updated using the credentials of the service object. See “LDAP Password Update Mode” on page 96 for more details. Applies only to LDAP service objects. |
| <code>--lookupcache-timeout</code> | The length of time, in seconds, for which LDAP lookup cache entries on the SGD server are held. See “Lookup Cache Timeout” on page 99 for more details. |
| <code>--ad-alwaysusegc</code> | Whether the global catalog is always be used for lookups. Enabling this option can speed up LDAP searches. See “Search Only the Global Catalog” on page 97 for more details. Applies only to Active Directory service objects. |
| <code>--suffix-mappings</code> | A list of mappings between domain names, used for Kerberos authentication. Each entry should be of the form <i>suffix=domain</i> , for example <code>test.east.example.com=east.example.com</code> . See “Suffix Mappings” on page 98 for more details. Applies only to Active Directory service objects. |
| <code>--file</code> | Specifies a file containing a batch of commands to edit service object attributes. |

Examples

The following example disables the `testldap` service object.

```
$ tarantella service edit --name testldap --enabled 0
```

The following example changes the position of the mainldap service object to third in the list of service objects.

```
$ tarantella service edit --name mainldap --position 3
```

tarantella service list

Lists the available service objects and their attributes.

See [“Using Service Objects” on page 90](#) for more details about service objects.

Syntax

```
tarantella service list { [ --name obj ]  
                        [ --setting... ]  
                        } | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|-----------|--|
| --name | Specifies the name of the service object to list. If no --name is specified, all service objects and their attribute values are listed. |
| --setting | Names a service object attribute you want to list the value for. If no --setting is specified, all attributes are listed for the service object. |
| --file | Specifies a file containing a batch of commands to list service objects and their attribute settings. |

Examples

The following example lists all service objects in their position order and their attribute values.

```
$ tarantella service list
```

The following example lists the values of the URL and Enabled attributes for the mainldap service object.

```
$ tarantella service list --name mainldap --url --enabled
```

The following example lists the all attribute values for the mainldap service object.

```
$ tarantella service list --name mainldap
```

tarantella service new

Creates a new service object.

By default, new service objects are enabled and added in the last position in the list of service objects.

See [“Using Service Objects” on page 90](#) for more details about service objects.

Syntax

```
tarantella service new {
    --name obj
    --type ldap|ad
    --url url...
    [ --position pos ]
    [ --enabled 0|1 ]
    [ --operation-timeout timeout ]
    [ --base-domain domain ]
    [ --default-domain domain ]
    [ --black-list list ]
    [ --white-list list ]
    [ --security-mode ""|clientcerts]
    [ --auth-mode kerberos|ssl ]
    [ --site-aware 0|1 ]
    [ --site-name name ]
    [ --check-pwd-policy 0|1 ]
    [ --pwd-expiry-warn-threshold threshold ]
    [ --pwd-expiry-fail-threshold threshold ]
    [ --domain-list domains ]
    [ --password-update-mode ldapuser|ldapadmin]
    [ --lookupcache-timeout timeout ]
    [ --ad-alwaysusegc 0|1 ]
    [ --suffix-mappings mappings ]
} | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|------------|--|
| --name | The name of the service object to create. See “Name” on page 447 for more details. |
| --type | The service object type, either LDAP or Active Directory. See “Type” on page 448 for more details. |
| --url | The URLs of the LDAP directories or the URL of an Active Directory forest. The URL(s) must be unique. Different service objects cannot use the same URL(s). See “URLs” on page 448 for more details. |
| --position | A number that specifies the position of the service object in the list of service objects. The number 1 means first position in the list. |

| Option | Description |
|----------------------------------|--|
| <code>--enabled</code> | Whether the service object is enabled for use. See “Enabled” on page 448 for more details. |
| <code>--operation-timeout</code> | Period of time, in seconds, to wait for a directory server to respond to an LDAP operation. See “LDAP Operation Timeout” on page 99 for more details. |
| <code>--base-domain</code> | The domain that SGD uses for Active Directory authentication if users only supply a partial domain when they log in. See “Active Directory Base Domain” on page 450 for more details. Applies only to Active Directory service objects. |
| <code>--default-domain</code> | The domain that SGD uses for Active Directory authentication if users do not supply a domain when they log in. See “Active Directory Default Domain” on page 450 for more details. Applies only to Active Directory service objects. |
| <code>--black-list</code> | A list of Active Directory servers which are never used for LDAP queries. See “Blacklists” on page 97 for more details. Applies only to Active Directory service objects. |
| <code>--white-list</code> | A list of Active Directory servers which are always used for LDAP queries. Servers not included in the list cannot be used. See “Whitelists” on page 97 for more details. Applies only to Active Directory service objects. |
| <code>--security-mode</code> | Whether client certificates are used to authenticate the SSL connection to an Active Directory server. This option is only used if <code>--auth-mode</code> is SSL. See “SSL Connections to Active Directory” on page 56 for more details. Applies only to Active Directory service objects. |
| <code>--auth-mode</code> | The mechanism used to secure the connection to an Active Directory server, either Kerberos or SSL. Kerberos is used by default. See “SSL Connections to Active Directory” on page 56 for more details. Applies only to Active Directory service objects. |
| <code>--site-aware</code> | Enables site awareness for the service object. If <code>--site-name</code> is not set, SGD attempts to discover site information automatically by contacting the global catalog. See “Sites” on page 96 for more details. Applies only to Active Directory service objects. |
| <code>--site-name</code> | A site name for the service object. This option is only used if <code>--site-aware</code> is enabled. See “Sites” on page 96 for more details. Applies only to Active Directory service objects. |

| Option | Description |
|--|--|
| <code>--check-pwd-policy</code> | Whether a user's password policy should be checked at authentication time. This option is used to enable LDAP password expiry features. See "Password Expiry" on page 95 for more details. |
| <code>--pwd-expiry-warn-threshold</code> | The period of time, in seconds, before password expiry where a warning message is shown on the webtop. See "Password Expiry" on page 95 for more details. |
| <code>--pwd-expiry-fail-threshold</code> | The period of time, in seconds, before password expiry where authentication is denied for a user and they are forced to update their password. See "Password Expiry" on page 95 for more details. |
| <code>--domain-list</code> | Defines a list of domains to be contacted when SGD starts. See "Domain Lists" on page 99 for more details. Applies only to Active Directory service objects. |
| <code>--password-update-mode</code> | Determines how aged passwords are handled. The default setting is <code>ldapuser</code> , meaning that passwords are updated using the authenticated user credentials. This results in a password <i>change</i> . A setting of <code>ldapadmin</code> means that passwords are updated using credentials of the service object. This results in a password <i>reset</i> . See "LDAP Password Update Mode" on page 96 for more details. Applies only to LDAP service objects. |
| <code>--lookupcache-timeout</code> | The length of time, in seconds, for which LDAP lookup cache entries on the SGD server are held. See "Lookup Cache Timeout" on page 99 for more details. |
| <code>--ad-alwaysusegc</code> | Whether the global catalog is always be used for lookups. Enabling this option can speed up LDAP searches. See "Search Only the Global Catalog" on page 97 for more details. Applies only to Active Directory service objects. |
| <code>--suffix-mappings</code> | A list of mappings between domain names, used for Kerberos authentication. Each entry should be of the form <code>suffix=domain</code> , for example <code>test.east.example.com=east.example.com</code> . See "Suffix Mappings" on page 98 for more details. Applies only to Active Directory service objects. |
| <code>--file</code> | Specifies a file containing a batch of commands to edit service object attributes. |

Examples

The following example creates an LDAP service object called `mainldap`. The service object is set to third position in the list of service objects and is enabled by default.

```
$ tarantella service new \  
--name mainldap --type ldap \  
--url "ldap://main1.example.com;ldap://main2.example.com" \  
--position 3
```

The following example creates an Active Directory service object called `east`. The service object is in the last position in the list of service objects by default, and is set to disabled.

```
$ tarantella service new \  
--name east --type ad \  
--url "ad://east.example.com" \  
--enabled 0
```

The `tarantella setup` Command

Enables you to change Setup options. Follow the instructions on your screen.

Syntax

```
tarantella setup
```

Description

You can turn weekly archiving on or off. If archiving is on, you can schedule the time when the log is created.

You can also choose to recreate the default objects and assigned applications links originally created at installation time. This does not remove any objects you have created, but it does replace any objects with the same names as the originals.

Examples

The following example enables you to change Setup options.

```
# tarantella setup
```

The tarantella start Command

Starts SGD services on the host.

Syntax

```
tarantella start [ --http | --https ] [ --servlet ]
```

```
tarantella start cdm | sgd | webserver [ --http | --https ]  
                                         [ --servlet ]
```

Description

If no subcommands are specified, this command starts both the SGD server and the SGD web server.

The following table shows the available options for this command.

| Option | Description |
|-----------|--|
| --http | Starts HTTP services (Apache). |
| --https | Starts HTTPS services (Apache). Requires a valid SSL certificate for the SGD web server. |
| --servlet | Starts Java Servlet extension and JSP technology services (Tomcat). |

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|------------|---|---|
| cdm | Starts <i>only</i> the UNIX platform client drive mapping services on the SGD server. | <code>"tarantella start cdm"</code> on page 828 |
| sgd | Starts <i>only</i> the SGD server. | <code>"tarantella start sgd"</code> on page 829 |
| webserver | Starts <i>only</i> the SGD web server. | <code>"tarantella start webserver"</code> on page 829 |

Note – All commands include a `--help` option. You can use `tarantella start subcommand --help` to get help on a specific command.

Examples

The following example starts the SGD server *and* HTTPS services on the SGD web server.

```
# tarantella start --https
```

```
tarantella start cdm
```

Starts only the UNIX platform client drive mapping (CDM) services on the SGD server.

Syntax

```
tarantella start cdm
```

Description

Starts UNIX platform CDM services on the SGD server where the command is run.

Examples

The following example starts UNIX platform CDM services on the SGD server.

```
# tarantella start cdm
```

```
tarantella start sgd
```

Starts only the SGD server.

Syntax

```
tarantella start sgd
```

Description

Starts the SGD server.

Examples

The following example starts the SGD server.

```
# tarantella start sgd
```

```
tarantella start webserver
```

Starts only the SGD web server.

Syntax

```
tarantella start webserver [ --http | --https ] [ --servlet ]
```

Description

If you do not use any command options, the command starts the SGD web server in both HTTP and HTTPS mode, providing valid SSL certificates are present on the host. If valid SSL certificates are not present, the command starts the SGD web server in HTTP mode only.

The following table shows the available options for this command.

| Option | Description |
|-----------|--|
| --http | Starts HTTP services (Apache). |
| --https | Starts HTTPS services (Apache). Requires a valid SSL certificate for the SGD web server. |
| --servlet | Starts Java Servlet extension and JSP technology services (Tomcat). |

Note – If you restart both Apache and Tomcat using separate subsequent commands, you must restart Tomcat first.

Examples

The following example starts Apache and Tomcat.

```
# tarantella start webserver
```

The tarantella status Command

Reports SGD server information.

Syntax

```
tarantella status
  [ --summary | --byserver | --server serv | --ping [serv]
    --originalstate ]
  [ --format text | xml ]
  [ --verbose ]
```

Description

Reports SGD server information, including array details, the number of user sessions and application sessions running or suspended across the array, and how those sessions are distributed.

The command indicates if there are any clock synchronization issues for the array. You use the `--byserver` option to show the clock setting on each server in the array.

The command indicates if the array is in a repaired state. You use the `--originalstate` option to list the members of the array before it was repaired.

If there are problems with the array, the command shows the following information:

- If the servers do not agree on the array membership, the output shows the array configuration as seen by every SGD server in the array.
- If there are any other errors, for example a Secret Key Identification (SKID) authentication error, the command reports the error it received from the problem server.

The following table shows the available options for this command.

| Option | Description |
|-------------------------|--|
| <code>--summary</code> | Summarizes the global information for the array. This is the default setting. |
| <code>--byserver</code> | Displays detailed information for each server in the array. |
| <code>--server</code> | Displays detailed information for the specified server. Type in a peer DNS name. |
| <code>--format</code> | Specifies the output format. The default setting is <code>text</code> . |

| Option | Description |
|-----------------|--|
| --ping | Performs a quick health check of all SGD servers in the array or a single specified SGD server. |
| --originalstate | If the server is in an array that is repaired, this option displays the original array members before it was repaired. |
| --verbose | Displays the server health check and lists servers being contacted, before generating the command output. |

Examples

The following example summarizes information about sessions across the array.

```
$ tarantella status
```

The following example reports detailed status information for the SGD server boston.indigo-insurance.com.

```
$ tarantella status --server boston.indigo-insurance.com
```

The following example reports the original status for a repaired SGD array.

```
$ tarantella status --originalstate
```

The tarantella stop Command

Stops SGD services on the SGD host, prompting if users are currently connected.

Syntax

```
tarantella stop [ --force | --kill ] [ --quiet ]
                [ --http | --https ] [ --servlet ]
```

```
tarantella stop cdm
```

```
tarantella stop sgd [ --force | --kill ] [ --quiet ]
```

```
tarantella stop webserver [ --http | --https ] [ --servlet ]
```

Description

If no subcommands are specified, this command stops both the SGD server and the SGD web server.



Caution – Never use the UNIX platform kill command to stop SGD services.

The following table shows the available options for this command.

| Option | Description |
|-----------|---|
| --force | Tries harder to stop SGD services. |
| --kill | Kills the process IDs used by SGD services. Only use this option if you are having difficulty stopping the SGD server by other means. |
| --quiet | Does not prompt. Stops SGD services even if users are connected. |
| --http | Stops HTTP services (Apache). |
| --https | Stops HTTPS services (Apache). Requires a valid SSL certificate for the SGD web server. |
| --servlet | Stops Java Servlet extension and JSP technology services (Tomcat). |

Stopping an SGD server causes all user sessions and application sessions, including suspended application sessions, to be terminated.

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|------------|---|---|
| cdm | Stops <i>only</i> the UNIX platform client drive mapping services on the SGD server | “tarantella stop cdm” on page 834 |
| sgd | Stops <i>only</i> the SGD server. | “tarantella stop sgd” on page 835 |
| webserver | Stops <i>only</i> the SGD web server. | “tarantella stop webserver” on page 835 |

Note – All commands include a `--help` option. You can use `tarantella stop subcommand --help` to get help on a specific command.

Examples

The following example stops the SGD server *and* the SGD web server, without displaying a confirmation message if users are currently connected.

```
# tarantella stop --quiet
```

```
tarantella stop cdm
```

Stops only the UNIX platform CDM services on the SGD server.

Syntax

```
tarantella stop cdm
```

Description

Stops UNIX platform CDM services on the SGD server where the command is run.

Examples

The following example stops UNIX platform CDM services on the SGD server.

```
# tarantella stop cdm
```

tarantella stop sgd

Stops SGD services on the SGD server.

Syntax

```
tarantella stop sgd [ --force | --kill ] [ --quiet ]
```

Description

Stops only the SGD server.

The following table shows the available options for this command.

| Option | Description |
|---------|---|
| --force | Tries harder to stop SGD services. |
| --kill | Kills the process IDs used by SGD services. Only use this option if you are having difficulty stopping the SGD server by other means. |
| --quiet | Does not prompt. Stops SGD services even if users are connected. |

Examples

The following example stops the SGD server.

```
# tarantella stop sgd
```

tarantella stop webserver

Stops only the SGD web server.

Syntax

```
tarantella stop webserver [ --http | --https ] [ --servlet ]
```

Description

If you do not use any command options, the command stops both the SGD web server and Tomcat services on the SGD host.

The following table shows the available options for this command.

| Option | Description |
|-----------|---|
| --http | Stops HTTP services (Apache). |
| --https | Stops HTTPS services (Apache). Requires a valid SSL certificate for the SGD web server. |
| --servlet | Stops Java Servlet extension and JSP technology services (Tomcat). |

Note – If you restart both Apache and Tomcat using separate subsequent commands, you must restart Tomcat first.

Examples

The following example stops Apache and Tomcat.

```
# tarantella stop webserver
```

The tarantella tokencache Command

This command manipulates the token cache used for logging in with an authentication token. SGD Administrators can list and delete entries in the token cache.

Syntax

```
tarantella tokencache delete | list
```

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|------------|--|--|
| delete | Deletes entries from the token cache. | “tarantella tokencache delete” on page 837 |
| list | Lists the contents of the token cache. | “tarantella tokencache list” on page 838 |

Note – All commands include a `--help` option. You can use `tarantella tokencache command --help` to get help on a specific command.

Examples

The following example deletes all entries in the token cache.

```
$ tarantella tokencache delete --all
```

The following example lists all entries in the token cache and the time the tokens were created.

```
$ tarantella tokencache list --creationtime
```

tarantella tokencache delete

Deletes entries in the token cache. The token cache is used for logging in with an authentication token.

Syntax

```
tarantella tokencache delete {  
    [ --username username | --all ]  
    [ --format text | xml ] }  
    | --file file
```

Description

The following table shows the available options for this command.

| Option | Description |
|------------|--|
| --username | Specifies the name of the entry to be deleted. |
| --all | Deletes all entries in the cache. |
| --format | Output format. The default setting is <code>text</code> . |
| --file | Specifies a batch file to process. The file contains one line per set of settings, each line using the above options. Use <code>--file -</code> to read from <code>stdin</code> . |

Examples

The following example deletes all entries in the token cache.

```
$ tarantella tokencache delete --all
```

tarantella tokencache list

Lists the contents of the token cache. The token cache is used for logging in with an authentication token.

Syntax

```
tarantella tokencache list [ --creationtime ]
                             [ --format text | xml ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------------|---|
| --creationtime | Lists the time each token in the cache was created. |
| --format | Specifies the output format. The default setting is text. |

Examples

The following example lists all entries in the token cache and the time the tokens were created.

```
$ tarantella tokencache list --creationtime
```



The tarantella tscal Command

Use the tarantella tscal command to manage Microsoft Windows Terminal Services CALs for non-Windows clients.

Syntax

| | | |
|------------------------------------|---------------------|-----------------------|
| <code>tarantella tscal free</code> | <code> list</code> | <code> return</code> |
|------------------------------------|---------------------|-----------------------|

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|------------|--|---|
| free | Frees a Terminal Services CAL for use by another non-Windows client. | “tarantella tscal free” on page 840 |
| list | Lists the Terminal Services CALs currently reserved for non-Windows clients. | “tarantella tscal list” on page 842 |
| return | Returns Terminal Services CALs to the Windows license server. | “tarantella tscal return” on page 843 |

Note – All commands include the `--help` option. You can use `tarantella tscal subcommand --help` to get help on a specific command.

Examples

The following example lists the Terminal Services CALs currently reserved for non-Windows clients.

| |
|---------------------------------------|
| <code>\$ tarantella tscal list</code> |
|---------------------------------------|

tarantella tscal free

Use the `tarantella tscal free` command to free a Microsoft Windows Terminal Services CAL so that it can be used by another non-Windows client.

Syntax

```
tarantella tscal free [ --inuseby user | --calid id ]
```

Description

You can only free a CAL if the user has no application sessions that use Windows Terminal Services.

Note – Freed CALs are not returned to the Windows license server.

Normally, you do not need to run this command, as SGD automatically frees a CAL as soon as a user exits their last Windows application. However, if an SGD server is removed from an array or it loses contact with the array, it might still be listed as using CALs. In this situation, you can run this command to free a CAL.

If you do not use any arguments, the command frees all CALs that have no application sessions that use Windows Terminal Services.

If you run this command on a secondary server in a SGD array and the primary server is unavailable, the CAL information might not be completely accurate. This is because the primary server is responsible for updating all SGD servers in the array with changes to CAL information. The command warns you if the primary is unavailable.

The following table shows the available options for this command.

| Option | Description |
|-----------|--|
| --inuseby | Free only the CALs for a particular user where the user is either of the following: <ul style="list-style-type: none">• The name of a user.• A wild card filter. The * character is the only character you can use in a wild card filter. It represents a string of any length containing any characters. So, an --inuseby " <i>*green*</i> " argument frees only the unused CALs for users whose name contains the string "green". |
| --calid | The ID of the CAL you want to free. Use the <code>tarantella tscal list</code> command to obtain the ID of the CAL you wish to free. |

Examples

The following example frees the CALs for Elizabeth Blue.

```
$ tarantella tscal free \  
--inuseby "o=Indigo Insurance/ou=Sales/cn=Elizabeth Blue"
```

tarantella tscal list

Use the `tarantella tscal list` command to list the Microsoft Windows Terminal Services CALs currently reserved for use by non-Windows clients.

Syntax

```
tarantella tscal list [ --inuseby user | --inuse | --free ]  
                      [ --type name ]  
                      [ --format text|xml ]
```

Description

If you do not use any arguments, the command lists all CALs and shows whether or not they are in use.

If you run this command on a secondary server in an SGD array and the primary server is unavailable, the list might not be completely accurate. This is because the primary server is responsible for updating all SGD servers in the array with changes to CAL information. The command warns you if the primary is unavailable.

The following table shows the available options for this command.

| Option | Description |
|-----------|---|
| --inuseby | List only the CALs being used by a particular user where the user is either of the following: <ul style="list-style-type: none">• The name of a user.• A wild card filter. You can use the <code>tarantella emulatorsession list</code> command to determine the name of a user. The * character is the only character you can use in a wild card filter. It represents a string of any length containing any characters. So, an <code>--inuseby "*"green"</code> argument lists only the CALs for users whose name contains the string "green". |
| --inuse | List only the CALs that are currently in use. |
| --free | List only the CALs that are currently not in use. |
| --type | List only the CALs that can connect to a particular type of Terminal Services server. This is either WinNT4-TS-CAL or Win200x-TS-CAL. Note - The name is not case sensitive. |
| --format | Specifies the output format. The default setting is <code>text</code> . |

Examples

The following example lists the CALs for non-Windows clients that are currently not in use.

```
$ tarantella tscal list --free
```

tarantella tscal return

Use the `tarantella tscal return` command to return all free Microsoft Windows Terminal Services CALs to the Windows license server.

Syntax

```
tarantella tscal return --free
```

Description

Note – The Windows license server might not reissue the returned CALs until approximately 90 days have elapsed since they were last in use.

Use the `tarantella tscal free` command to free a CAL so that it can be returned.

Normally, you do not need to run this command, as SGD automatically returns a CAL if it has not been used for 90 days. However, if an SGD server is removed from an array, you can use this command to manually return the CALs.

The following table shows the available options for this command.

| Option | Description |
|---------------------|--|
| <code>--free</code> | Returns all free CALs to the Windows license server. |

Examples

The following example returns all free CALs to the Windows license server.

```
$ tarantella tscal return --free
```

The `tarantella uninstall` Command

Uninstalls SGD or the specified SGD packages.

Syntax

```
tarantella uninstall { [ package... ] [ --purge ] | --list }
```

Description

Removes SGD or parts of it from your system, or lists the installed SGD packages.

If the SGD server is a member of an array, remove it from the array before you run `tarantella uninstall`. You can use the `tarantella array` command to do this.

The following table shows the available options for this command.

| Option | Description |
|-------------------|--|
| <i>package...</i> | Specifies individual packages to uninstall. If no packages are specified, the command uninstalls all SGD packages. SGD currently installs as a single package. |
| --purge | If all SGD packages are removed, this option also removes all configuration information related to your organization. If --purge is omitted, configuration information is left intact. |
| --list | Lists all SGD packages currently installed. |

Examples

The following example completely uninstalls SGD, removing all configuration information.

```
# tarantella uninstall --purge
```

The `tarantella version` Command

Reports the version numbers of installed SGD components.

Syntax

```
tarantella version
```

Description

Displays the version numbers of SGD components installed on the SGD server, together with information about the SGD server.

Information about installed SGD components is also available on the webtop. Click the ? button, in the lower-left corner of the webtop.

Examples

The following example displays the version numbers of installed SGD components.

```
$ tarantella version
```

The `tarantella webserver` Command

Use the `tarantella webserver` command to configure trusted users for the third-party authentication mechanism.

Syntax

```
tarantella webserver add_trusted_user | delete_trusted_user |  
list_trusted_users
```

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|---------------------|---|--|
| add_trusted_user | Adds the user name and password of a user that is to be trusted by the third-party authentication mechanism. | “tarantella webserver add_trusted_user” on page 847 |
| delete_trusted_user | Deletes the user name and password of a user that is to be trusted by the third-party authentication mechanism. | “tarantella webserver delete_trusted_user” on page 848 |
| list_trusted_users | Lists the user names of the users that are to be trusted by the third-party authentication mechanism. | “tarantella webserver list_trusted_users” on page 849 |

Note – All commands include the `--help` option. You can use `tarantella webserver subcommand --help` to get help on a specific command.

Examples

The following example lists trusted users.

```
# tarantella webserver list_trusted_users
```

```
tarantella webserver add_trusted_user
```

Adds the user name and password of a user that is to be trusted for third-party authentication.

Syntax

```
tarantella webserver add_trusted_user username
```

Description

After you enter the *username*, SGD prompts you to enter the password. The password must be at least six characters long.

You must restart the SGD web server, using `tarantella restart webserver`, to activate the new user.

You cannot use this command to change the password of a trusted user. You must delete the trusted user first, using `tarantella webserver delete_trusted_user`.

This command adds the user name to the “database” of Tomcat users in `/opt/tarantella/webserver/tomcat/tomcat-version/conf/tomcat-users.xml` and creates an SHA digest of the password. The user is also assigned the “SGDExternalAuth” role. This role is required to access the SGD external authentication web service.

Examples

The following example adds L3nNy_G0db3r as a trusted user.

```
# tarantella webserver add_trusted_user L3nNy_G0db3r
```

```
tarantella webserver  
delete_trusted_user
```

Deletes the user name and password of a user that is to be trusted for third-party authentication.

Syntax

```
tarantella webserver delete_trusted_user username
```

Description

You must restart the SGD web server, using `tarantella restart webserver`, to deactivate the user.

This command removes the specified user name from the “database” of Tomcat users in `/opt/tarantella/webserver/tomcat/tomcat-version/conf/tomcat-users.xml`.

Examples

The following example deletes L3nNy_G0db3r as a trusted user.

```
# tarantella webserver delete_trusted_user L3nNy_G0db3r
```

```
tarantella webserver  
list_trusted_users
```

Lists the user names of the users that are to be trusted for third-party authentication.

Syntax

```
tarantella webserver list_trusted_users
```

Description

Each user name is separated by a comma. The command also shows whether or not the third-party authentication is currently enabled.

This command lists the user names in the “database” of Tomcat users in `/opt/tarantella/webserver/tomcat/tomcat-version/conf/tomcat-users.xml`.

Examples

The following example lists trusted users.

```
# tarantella webserver list_trusted_users
```

The tarantella webtopsession Command

This command enables SGD Administrators to list and end user sessions.

Syntax

```
tarantella webtopsession list | logout
```

Description

The following table shows the available subcommands for this command.

| Subcommand | Description | More Information |
|------------|--|---|
| list | Lists user sessions matching the person or server specified. | “tarantella webtopsession list” on page 851 |
| logout | Logs users out of SGD. | “tarantella webtopsession logout” on page 852 |

Note – All commands include a `--help` option. You can use `tarantella webtopsession subcommand --help` to get help on a specific command.

Examples

The following example displays details of all user sessions maintained by the SGD server detroit.

```
$ tarantella webtopsession list \  
--server "o=Indigo Insurance/cn=detroit"
```

The following example logs out Emma Rald from SGD.

```
$ tarantella webtopsession logout \  
--person "o=Indigo Insurance/ou=Marketing/cn=Emma Rald"
```

tarantella webtopsession list

Lists user sessions matching the person or server specified.

Syntax

```
tarantella webtopsession list [ --person pobj | --server serv ]  
                             [ --format text|count|xml ]
```

Description

For each session, the following details are displayed:

- **Print state.** Shows whether the user has paused printing or not.
- **Client.** The IP address of the client.
- **Logged in at.** The timestamp when the user logged in.
- **User.** The name of the user.
- **Logged in to.** The SGD server hosting the user session.
- **Connection type.** Whether the connection is a standard or a secure connection.

You can list user session details using the following Administration Console tabs:

- Sessions tab
- Secure Global Desktop Servers → User Sessions tab
- User Sessions tab for a user profile object

The following table shows the available options for this command.

| Option | Description |
|----------|---|
| --person | Displays details of user sessions matching the person specified. Use the user identity of the person. |
| --server | Displays details of user sessions matching the SGD server specified. Use the name of an application server object in the local repository (if the SGD server has an application server object), or the peer DNS name for the server. |
| --format | Specifies the output format. The default setting is text. Use count to display only the number of matching sessions. |

If neither a person nor server is specified, the command lists all user sessions across the array.

Guest users and anonymous users have unique names, even though they can share the same profile in the System Objects organization. To name a guest or anonymous user, use the unique name and not the name of the profile object.

For example, `.../_dns/newyork.indigo-insurance.com/_anon/1`.

Note – Make sure you quote any object names containing spaces, for example, "`o=Indigo Insurance`".

Examples

The following example displays details of all user sessions maintained by the SGD server `detroit`.

```
$ tarantella webtopsession list \  
--server "o=Indigo Insurance/cn=detroit"
```

The following example displays all user sessions across the array.

```
$ tarantella webtopsession list
```

tarantella webtopsession logout

Ends the user session for each person specified. This has the effect of logging them out of SGD.

Syntax

```
tarantella webtopsession logout --person pobj...
[ --format text|quiet ]
```

Description

The following table shows the available options for this command.

| Option | Description |
|----------|--|
| --person | Ends the user session of the specified person. Use the user identity of the person. |
| --format | Specifies the output format. The default setting is <code>text</code> . With <code>--format quiet</code> , no messages are displayed and the exit code indicates the number of sessions logged out. |

You can end user sessions using the following Administration Console tabs:

- Sessions tab
- Secure Global Desktop Servers → User Sessions tab
- User Sessions tab for a user profile object

Guest users and anonymous users have unique names, even though they can share the same profile in the System Objects organization. To name a guest or anonymous user, use the unique name and not the name of the profile object.

For example, `.../_dns/newyork.indigo-insurance.com/_anon/1`.

Note – Make sure you quote any object names containing spaces, for example, `"o=Indigo Insurance"`.

Examples

The following example logs out Emma Rald from SGD.

```
$ tarantella webtopsession logout \  
--person "o=Indigo Insurance/ou=Marketing/cn=Emma Rald"
```

The following example ends an anonymous user's user session.

```
$ tarantella webtopsession logout \  
--person ../_dns/newyork.indigo-insurance.com/_anon/1
```

Login Scripts

This appendix contains reference information about the Oracle Secure Global Desktop (SGD) login scripts. You can use this information to customize the standard SGD login scripts, or to develop your own login scripts.

This appendix includes the following topics:

- “Login Scripts Supplied With SGD” on page 855
- “Login Script Tcl Commands and Procedures” on page 858
- “Login Script Variables” on page 866
- “Login Script Timeouts” on page 874
- “Login Script Error Messages” on page 878

Login Scripts Supplied With SGD

All login scripts supplied with SGD are stored in the `/opt/tarantella/var/serverresources/expect` directory.

SGD login scripts are written in Tcl (version 8.4) and Expect (version 5.43). Expect extends Tcl and provides additional commands for interacting with programs.

For more information about Tcl, see the Tcl Developer Exchange (<http://www.tcl.tk>).

For more information about Expect, see The Expect Home Page (<http://expect.nist.gov/>).

The login scripts can be divided into the scripts that can be used when configuring applications and the scripts that contain common code. The available scripts are described in the following sections.

Login Scripts Used When Configuring Applications

You configure the login script that is used for an application as follows:

- In the Administration Console, use the Login Script script attribute on the Launch tab for the application object.
- On the command line, use the `--login script` command option with the `tarantella object` commands.

The following table lists the login scripts supplied with SGD that you can set as the Login Script attribute for an application object, and a description of what they are used for.

| Script Name | Description |
|---------------------------|---|
| <code>unix.exp</code> | The standard login script for character and X applications. If the Login Script attribute is blank, this script is used by default. Can be used with all application Connection Methods. |
| <code>securid.exp</code> | Replacement for <code>unix.exp</code> if you are using SecurID for application server authentication. See “Using RSA SecurID for Application Authentication” on page 189. |
| <code>windows.exp</code> | The standard login script for Windows applications If the Login Script attribute is blank, this script is used by default. |
| <code>3270.exp</code> | The standard login script for 3270 applications. If the Login Script attribute is blank, this script is used by default. Can be used with all application Connection Methods. The script builds a command to run the TeemTalk for UNIX platform terminal emulation software. |
| <code>5250.exp</code> | The standard login script for 5250 applications. If the Login Script attribute is blank, this script is used by default. Can be used with all application Connection Methods. The script builds a command to run the TeemTalk for UNIX platform terminal emulation software. |
| <code>vms.exp</code> | Used for X or character applications running on Virtual Memory System (VMS) application servers. Can be used for all application Connection Methods except <code>rexec</code> . See “Configuring VMS Applications” on page 201. |
| <code>vmsrexec.exp</code> | Used for X or character applications running on VMS application servers. Can only be used if the application Connection Method is <code>rexec</code> . See “Configuring VMS Applications” on page 201. |

| Script Name | Description |
|----------------------------|---|
| <code>unixclass.exp</code> | Script used to create a shadowable UNIX [®] platform session, for use in a virtual classroom situation. See “Creating a Virtual Classroom” on page 196. |
| <code>winclass.exp</code> | Script used to create a shadowable Windows session, for use in a virtual classroom situation. See “Creating a Virtual Classroom” on page 196. |
| <code>pupil.exp</code> | Script used by the pupils when shadowing a teacher in a virtual classroom situation. See “Creating a Virtual Classroom” on page 196. |

Login Scripts Containing Common Code

The following table lists the login scripts supplied with SGD that contain common code and a description of what they are used for. These scripts must not be set as the Login Script attribute for an application object.

| Script Name | Description |
|-------------------------------|---|
| <code>runsubscript.exp</code> | The standard wrapper login script used to call all the other SGD login scripts. Sets the environment variables the login scripts are allowed to use. |
| <code>procs.exp</code> | Called by other scripts. Defines common Tcl procedures. |
| <code>vars.exp</code> | Called by other scripts. Defines the variables, messages, and timeouts used by the other login scripts. |
| <code>securid-vars.exp</code> | Called by <code>securid.exp</code> . Defines additional variables and messages needed for SecurID authentication. |
| <code>xauth.exp</code> | Called by <code>procs.exp</code> and <code>classroom.exp</code> . Used to handle the X authorization process, including the X authorization permissions for shadowing. |
| <code>classroom.exp</code> | Called by <code>unixclass.exp</code> , <code>winclass.exp</code> , and <code>pupil.exp</code> . Defines common procedures for retrieving the X display to shadow. |

| Script Name | Description |
|----------------------------|--|
| <code>unixwin.exp</code> | Used for Windows applications configured to use the Citrix Independent Computing Architecture (ICA) protocol. This script assumes that the user's <code>PATH</code> includes the directory where the Merge or ICA UNIX client software is installed. Although SGD no longer supports the SCO Merge or Citrix ICA protocols, legacy Windows application objects can continue to use them. |
| <code>wcpwts.exp</code> | Called by <code>windows.exp</code> for Windows applications configured to use the Microsoft Remote Desktop Protocol (RDP) protocol. |
| <code>wincenter.exp</code> | Used for Windows applications configured to use the WinCenter or Citrix UNIX Integration Services protocol. Although SGD no longer supports the WinCenter and Citrix UNIX Integration Services protocols, legacy Windows application objects can continue to use them. |

Login Script Tcl Commands and Procedures

The login scripts supplied with SGD use several Tcl commands and procedures for communication with the application server.

The *Tcl commands* are commands that are defined in the Execution Protocol Engine component of SGD. The commands can be used in your own login scripts to provide control over the connection to the application server, and the display of the SGD Application Authentication and Progress dialogs.

The *Tcl procedures* are defined in the login scripts only. The procedures can be used to provide more control over the SGD Application Authentication dialog.

Controlling the SGD Application Authentication Dialog

The following Tcl commands and procedures are used to control the display of the SGD Application Authentication dialog when starting applications:

- [“authrequest” on page 859](#)
- [“authenticate” on page 860](#)
- [“authenticate2” on page 860](#)

- “customauthenticate” on page 861
- “userauthenticate” on page 861

authrequest

```
authrequest
[ -normal | -changed ]
-showuser 0|1
-title title
-message message
-customuserlabel 0|1
-userlabel label
-custompasswdlabel 0|1
-passwdlabel label
-showpasscache 0|1
-showsmartcard 0|1
-isuserdialog 0|1|2
```

This Tcl command displays a dialog box that indicates a problem with the user name or password.

Typically you do not call the `authrequest` command directly in your login scripts. Instead, you use the defined Tcl procedures to call this command with the required arguments.

This command has the following arguments.

| Argument | Description |
|--------------------|---|
| -normal | Specifies that the password is incorrect. |
| -changed | Specifies that the password has expired. |
| -showuser | Specifies that the user name field is displayed. |
| -showpasswd | Specifies that the password field is displayed. |
| -title | Specifies the title used for the authentication dialog. |
| -message | Specifies the message that is displayed in the authentication dialog. |
| -customuserlabel | Specifies whether to use a customized label for the user name field. |
| -userlabel | Specifies the customized label to use for the user name field. |
| -custompasswdlabel | Specifies whether to use a customized label for the password field. |
| -passwdlabel | Specifies the customized label to use for the password field. |

| Argument | Description |
|---------------------|---|
| -showpasscache | Specifies whether the Save This Password check box is displayed. |
| -showsmartcard | Specifies whether the smart card option is displayed. |
| -isuserdialog 0 1 2 | Specifies whether a customized authentication dialog is used. Specify one of the following: <ul style="list-style-type: none"> • 0 - Do not check whether the user name has changed • 1 - Check whether the user name has changed and reconnect to the application server if necessary • 2 - Use the default SGD behavior if the user name has changed |

The following example displays a dialog box that says the password is incorrect.

```
authrequest -normal
```

authenticate

Displays a dialog box that indicates a problem with the user name or password.

This Tcl procedure calls the `authrequest` command with the following options.

```
authenticate [ -normal | -changed ]
```

authenticate2

Displays a dialog box that indicates a problem with the user name or password. You can use your own title for the dialog and display your own message. You can also control whether the user name and password fields display.

This Tcl procedure calls the `authrequest` command with the following arguments.

```
authenticate2
[ -normal | -changed ]
-showuser 0|1
-showpasswd 0|
-title title
-message message
```

customauthenticate

Displays a dialog box that indicates a problem with the user name or password. You can fully customize the authentication dialog.

With this procedure, the Execution Protocol Engine does not check the text the user types in the user name field. If Secure Shell (SSH) is used as the connection method for the application and the user changes the user name, the Execution Protocol Engine does not break the connection and reconnect as the new user name. This can cause applications to fail to start. If you are using SSH and allow the user to change the user name, use the `userauthenticate` procedure instead.

This Tcl procedure calls the `authrequest` command with the following arguments.

```
customauthenticate
[ -normal | -changed ]
-showuser 0|1
-title title
-message message
-customuserlabel 0|1
-userlabel label
-custompasswdlabel 0|1
-passwdlabel label
-showpasscache 0|1
-showsmartcard 0|1
```

userauthenticate

Displays a dialog box that indicates a problem with the user name or password. You can fully customize the authentication dialog.

This procedure is the same as `customauthenticate` except that it does check whether the user has changed the user name. If the user name is changed, the Execution Protocol Engine breaks the connection to the application server and reconnects as the changed user.

This Tcl procedure calls the `authrequest` command with the following arguments:

```
userauthenticate
[ -normal | -changed ]
-showuser 0|1
-showpasswd 0|1
-title title
-message message
-customuserlabel 0|1
-userlabel label
-custompasswdlabel 0|1
```

```
-passwdlabel label
-showpasscache 0|1
-showsmartcard 0|1
```

Controlling the SGD Progress Dialog

The following Tcl commands are used to control the display of the SGD progress dialog when starting applications:

- [“loadererror” on page 862](#)
- [“clienttimer” on page 862](#)
- [“canceltimer” on page 862](#)
- [“progress” on page 863](#)

loadererror

```
loadererror error
```

You can use this Tcl command to override the error message returned by the login script. You can use this function, for example, to replace the standard login script error messages with your own message. If the application fails to start, the error is displayed in the progress dialog and in the log files. See [“Login Script Error Messages” on page 878](#).

clienttimer

```
clienttimer [ time ] [ message ] [ timers ]
```

This Tcl command displays *message* in the progress dialog box for the specified *time*. The progress bar has *timers* sections in total. The following is an example.

```
clienttimer 10 "Launching the application" 4
```

canceltimer

```
canceltimer
```

This Tcl command cancels the `clienttimer` command. This command has no arguments.

progress

```
progress [ message ]
```

This Tcl command displays *message* in the progress dialog box. The following is an example.

```
progress "Initializing..."
```

Controlling the Connection to the Application Server

The following Tcl commands are used to control the connection to the application server:

- [“setbuffer” on page 863](#)
- [“locallaunch” on page 864](#)
- [“tarantella” on page 865](#)
- [“sgdconnect” on page 865](#)

setbuffer

```
setbuffer [ -buffer num ] [ -output 0|1 ]
```

This Tcl command defines the number of bytes to read from the application server.

| Argument | Description |
|--------------------|---|
| -buffer <i>num</i> | Specifies the number of bytes. Default is 1. |
| -output 0 1 | Turns output on (1) or off (0). Default is 1. |

The following is an example.

```
setbuffer -buffer 1000
```

locallaunch

```
locallaunch [ -start ] [ -abort ] [ -user launchspec -root launchspec ]
```

This Tcl command starts an application when the application server is also the SGD server. This is known as an *optimized launch*.

| Argument | Description |
|-------------------------|---|
| -start | Starts an optimized launch. |
| -abort | Stops the optimized launch and reverts to the standard connection method. |
| -user <i>launchspec</i> | <p>Defines the connection methods to use for starting applications on the SGD server when the user is not the UNIX or Linux platform root user. You can specify different behavior for applications that are detached on launch, background applications, and those that are not, foreground applications.</p> <p>The <i>launchspec</i> can be one of the following:</p> <ul style="list-style-type: none">• 0 - Starts all applications using the Connection Method defined for the application object• 1 - Background applications use /bin/su. Foreground applications use the application object's Connection Method• 2 - Background applications use the application object's Connection Method. Foreground applications use /bin/su• 3 - Starts all applications using /bin/su <p>The default is 1.</p> |
| -root <i>launchspec</i> | <p>Defines the connection methods to use for starting applications on the SGD server when the user is the UNIX or Linux platform root user. You can specify different behavior for applications that are detached on launch, background applications, and those that are not, foreground applications.</p> <p>The <i>launchspec</i> can be one of the following:</p> <ul style="list-style-type: none">• 0 - Starts all applications using the Connection Method defined for the application object• 1 - Background applications use /bin/su. Foreground applications use the application object's Connection Method• 2 - Background applications use the application object's Connection Method. Foreground applications use /bin/su• 3 - Starts all applications using /bin/su• 4 - Starts all applications using the Connection Method defined for the application object <p>The default is 3.</p> |

The following is an example.

```
locallaunch -abort
```

tarantella

```
tarantella -nosocket -portnumber num -thirdtiershell shell
```

This Tcl command is used to configure the connection to the application server. You must configure the connection before you use the `sgdconnect` command.

| Argument | Description |
|---|--|
| <code>-nosocket</code> | Specifies that the application is to be started by some other means and must be implemented by whoever is creating the script, for example by using Expect's <code>spawn</code> command. This can only be done with applications that do not require a permanent connection, such as X applications. This command is useful if you have an unusual application server, or if you need to integrate with an existing application start mechanism. |
| <code>-portnumber <i>num</i></code> | Overrides the port used to make the connection to the application server. If you use this option, you must execute the <code>tarantella</code> command before the <code>sgdconnect</code> command, otherwise the port number is ignored. |
| <code>-thirdtiershell <i>shell</i></code> | Specifies the shell to use on the application server, for example <code>/bin/sh</code> . |

The following example connects to the application server on TCP port 5999.

```
tarantella -portnumber 5999
```

sgdconnect

```
sgdconnect
```

Instructs the Execution Protocol Engine to connect to the application server. This command has no arguments.

Most of the SGD login scripts use `sgdconnect` to make the connection. If you want to handle the connection to the application server yourself, you must ensure that your script does not use this command.

The `wcpwts.exp` standard login script is an example of a login script that does not use this command to connect to an application server.

Login Script Variables

SGD login scripts use and support a number of variables. The variables can be divided into guaranteed variables, that are always available, and optional variables, that are only available if they have a value.

To be able to use a variable in a login script, it must be defined in the `runsubscript.exp` login script.

The following sections list the guaranteed and optional variables, and a description of their purpose.

Guaranteed Login Script Variables

Guaranteed variables store the names of commands to run, the application server to log in to, and the connection method to use.

All login scripts use at least some of the guaranteed variables.

Guaranteed variables always exist, though they might have a null value.

| Variable | Description |
|--------------------------|--|
| ALTDISPLAY | The fully qualified Domain Name System (DNS) name of the user's client device and the display number being used. |
| DISPLAY | The Internet Protocol (IP) address of the user's client device and the display number being used. |
| TTA_AGEDPASSWORD | Whether to use the manual or dialog method of dealing with aged passwords. |
| TTA_ALLOWTHIRDTIERDIALOG | Whether to show a dialog box on the application server if the user's password is aged, missing or incorrect. This variable can have the following values: <ul style="list-style-type: none">• <code>user</code> - If the user holds down the Shift key when they click an application's link or if there is a password problem• <code>system</code> - Only when there is a password problem• <code>none</code> - Never show a dialog box |
| TTA_AUXCOMMANDS | Any auxiliary commands to run on the application server. This corresponds to the application object's Window Manager attribute. |

| Variable | Description |
|-------------------------|--|
| TTA_CLIENT_IPADDR | The IP address of the user's client device. This is the IP address obtained by the SGD Client. |
| TTA_COMMAND | The command to run on the application server. This corresponds to the application object's Application Command attribute. |
| TTA_CONNECTIONSERVICE | The transport used to connect to the application server. This corresponds to the application object's Connection Method attribute. |
| TTA_ENVIRONMENT | Any environment variable settings required on the application server. This corresponds to the application object's Environment Variables attribute. |
| TTA_HOSTNAME | The application server that the login script connects to. This is chosen by application load balancing, from those listed on the Hosting Application Servers tab for the application object. |
| TTA_HOSTPROBE | The path to the ttahostprobe binary. Used to check whether an application server is available. |
| TTA_IPADDRESS | The application server's IP address. |
| TTA_LOGFILE | <p>The name of a file where error and diagnostic messages are logged. By default, this has the form <i>scriptID.log</i>, where <i>script</i> is the name of the login script and <i>ID</i> is its process ID on the SGD server.</p> <p>If set to null, messages are not stored.</p> <p>To log messages in this file, include the following code in your login script:</p> <pre>log_file \$env(TTA_LOGFILE)</pre> |
| TTA_PORT | The port used to connect to the application server for the Connection Method configured for the application object. |
| TTA_PRIMARY_DNSNAME | <p>The primary SGD server's fully qualified DNS name.</p> <p>This lets the login script choose the correct SGD printer when setting the default printer value. It is used to differentiate between multiple entries in the <i>/etc/ttaprinter.conf</i> file.</p> |
| TTA_SCRIPT | The Expect script that runs after <i>runsubscript.exp</i> , for example <i>unix.exp</i> . |
| TTA_SECOND_TIER_DNSNAME | <p>The fully qualified DNS name of the SGD server hosting the application session.</p> <p>Used with <i>TTA_THIRD_TIER_DNSNAME</i> to determine whether the application server and the SGD server are the same, and use an optimized launch process if they are.</p> |
| TTA_THIRD_TIER_DNSNAME | <p>The fully qualified DNS name of the application server hosting the application.</p> <p>Used with <i>TTA_SECOND_TIER_DNSNAME</i> to determine whether the application server and the SGD server are the same, and use an optimized launch process if it they are.</p> |
| TTA_THIRD_TIER_VARS | The list of variables to export to the environment on the application server. |

| Variable | Description |
|--------------------|--|
| TTA_STDERR | A temporary error file. |
| TTA_WILLDISCONNECT | Whether the connection is broken once the command is executed. |
| TTA_XLAUNCH | Whether the application is an X application. The value of this variable is 0 or 1. |

The following guaranteed variables are also defined in `runsubscript.exp`. These are variables used by the SGD server when starting applications:

- LANG
- LANGUAGE
- LC_ALL
- LC_CTYPE
- LC_NUMERIC
- LC_TIME
- LC_COLLATE
- LC_MONETARY
- LC_MESSAGES
- LC_PAPER
- LC_NAME
- LC_ADDRESS
- LC_TELEPHONE
- LC_MEASUREMENT
- LC_IDENTIFICATION
- PATH
- TTA_PreferredLocale
- TTABASEDATADIR
- TTADATADIR
- TTADIR

Optional Login Script Variables

Optional variables store additional information about the application, the user, and their session.

Optional variables are often used to test conditions and modify the login script's behavior accordingly. Optional variables only exist if they have a value. For example, the `TTA_ResumeTimeOut` variable only exists if the application object's `Application Resumability` attribute has a value.

Most optional variables contain the values of object attributes. The application being started has its application object's attributes made available as optional variables. Similarly, the attributes of the user profile are also made available in the same way. Other optional variables contain additional information about the user's session.

| Variable | Description |
|---------------------------------------|---|
| <code>TTA_AdminMode</code> | Corresponds to the Windows application object's <code>Console Mode</code> attribute. |
| <code>TTA_Appearance</code> | Corresponds to a character application object's <code>Border Style</code> attribute. |
| <code>TTA_AppletHeight</code> | Corresponds to the application object's <code>Window Size: Height</code> attribute. |
| <code>TTA_AppletWidth</code> | Corresponds to the application object's <code>Window Size: Width</code> attribute. |
| <code>TTA_ApplicationName</code> | The application object's fully qualified name. |
| <code>TTA_ApplicationPlacement</code> | Corresponds to the application object's <code>Window Type</code> attribute. This variable can have the value <code>multiplewindows</code> - client window management, <code>awtwindow</code> - independent window, <code>kiosk</code> - kiosk, <code>localx</code> - local X server, and <code>seamlesswindows</code> - seamless window. |
| <code>TTA_Arguments</code> | Corresponds to the application object's <code>Arguments For Command</code> attribute. |
| <code>TTA_AudioQuality</code> | Corresponds to the Windows <code>Audio Quality</code> attribute on the <code>Global Settings → Client Device</code> tab in the Administration Console. This variable can have the value <code>low</code> , <code>medium</code> , and <code>high</code> . |
| <code>TTA_Autowrap</code> | Corresponds to the character application object's <code>Line Wrapping</code> attribute. |
| <code>TTA_BackgroundColor</code> | Corresponds to the <code>Background Color</code> attribute for a 3270 or 5250 application object. |
| <code>TTA_ButtonLevels</code> | Corresponds to the 3270 or 5250 application object's <code>Displayed Soft Buttons</code> attribute. This variable can have the value 0 - No Buttons), 1 - 2 Rows, 2 - 4 Rows, 3 - 6 Rows, and 4 - 8 Rows. |
| <code>TTA_CachePassword</code> | Whether the user selected the <code>Save This Password?</code> box when they supplied a user name and password for the application server. |
| <code>TTA_CodePage</code> | Corresponds to the character application object's <code>Code Page</code> attribute. This variable can have the value 437, 850, 852, 860, 863, 865, 8859-1, 8859-2, Multinational, Mazovia, or CP852. |
| <code>TTA_ColorMap</code> | Corresponds to the character application object's <code>Color Map</code> attribute. |

| Variable | Description |
|---------------------------|--|
| TTA_Columns | Corresponds to the character application object's Window Size: Columns attribute. |
| TTA_Compression | Corresponds to the application object's Command Compression attribute. This variable can have the value <i>automatic</i> , <i>on</i> , or <i>off</i> . |
| TTA_ContinuousMode | Corresponds to the application object's Command Execution attribute. This variable can have the value <i>automatic</i> , <i>on</i> , or <i>off</i> . |
| TTA_ControlCode | Corresponds to the character application object's Escape Sequences attribute. This variable can have the value <i>7-bit</i> , or <i>8-bit</i> . |
| TTA_Cursor | Corresponds to the character application object's Cursor attribute. This variable can have the value <i>off</i> , <i>block</i> , or <i>underline</i> . |
| TTA_CursorKeyMode | Corresponds to the character application object's Cursor Key Codes Modification attribute. This variable can have the value <i>application</i> , or <i>cursor</i> . |
| TTA_DelayedUpdate | Corresponds to the application object's Delayed Updates attribute. |
| TTA_DisableCursorSettings | Corresponds to the Windows application object's Cursor Settings attribute. |
| TTA_DisableCursorShadow | Corresponds to the Windows application object's Cursor Shadow attribute. |
| TTA_DisableFullWindowDrag | Corresponds to the Windows application object's Full Window Drag attribute. |
| TTA_DisableMenuAnimations | Corresponds to the Windows application object's Menu Animations attribute. |
| TTA_DisableTheming | Corresponds to the Windows application object's Theming attribute. |
| TTA_DisableWallpaper | Corresponds to the Windows application object's Desktop Wallpaper attribute. |
| TTA_DisplayEnginePage | Corresponds to the application object's Emulator Applet Page attribute. Note - This attribute is no longer used. |
| TTA_DisplayName | Corresponds to the user profile's Name attribute. |
| TTA_Domain | Corresponds to the application object's Domain Name attribute. |
| TTA_EnableFontSmoothing | Corresponds to the Windows application object's Font Smoothing attribute. |
| TTA_EuroMapping | Corresponds to the application object's Euro Character attribute. This variable can have the value <i>iso8859-15</i> , or <i>unicode</i> . |
| TTA_FilePath | Corresponds to the application object's Application Command attribute. |

| Variable | Description |
|--------------------------|---|
| TTA_FixedFontSize | Corresponds to the character application object's Font Size: Fixed attribute. |
| TTA_FontFamily | Corresponds to the character application object's Font Family attribute. This variable can have the value <code>courier</code> , <code>helvetica</code> , or <code>timesroman</code> . |
| TTA_FontSize | Corresponds to the character application object's Font Size attribute. |
| TTA_ForegroundColor | Corresponds to the 3270 or 5250 application object's Foreground Color attribute. |
| TTA_GraphicsAcceleration | Corresponds to the application object's Graphics Acceleration attribute. |
| TTA_Height | Corresponds to the application object's Window Size: Height attribute. This variable provides the same information as <code>TTA_AppletHeight</code> . |
| TTA_HostLocale | Corresponds to the application server object's Prompt Locale attribute. |
| TTA_HostName | The application server that the login script connects to. This is chosen by application server load balancing, from those listed on the Hosting Application Servers tab for the application object. |
| TTA_IBMHostName | Corresponds to the 3270 or 5250 application object's Server Address attribute. |
| TTA_Icon | Corresponds to the application object's Icon attribute. |
| TTA_InstanceName | The application session ID. |
| TTA_InterlacedImages | Corresponds to the application object's Interlaced Images attribute. This variable can have the value <code>automatic</code> , <code>on</code> , or <code>off</code> . |
| TTA_KeyboardType | Corresponds to the 3270 or 5250 application object's Keyboard Type attribute. This variable can have the value <code>pc</code> , <code>sun4</code> , <code>sun5</code> , and <code>hp</code> . |
| TTA_KeymapLock | Corresponds to the application object's Keyboard Map attribute. |
| TTA_KeypadMode | Corresponds to the character application object's Numpad Codes Modification attribute. This variable can have the value <code>application</code> , or <code>numeric</code> . |
| TTA_Lines | Corresponds to the character application object's Window Size: Lines attribute. |
| TTA_LocalAddr | The IP address of the SGD host. |
| TTA_LoginScript | Corresponds to the application object's Login Script attribute. |
| TTA_Maximise | Corresponds to the 3270 or 5250 application object's Window Size attribute. |
| TTA_MiddleMouseTimeout | Corresponds to the application object's Middle Mouse Timeout attribute. |

| Variable | Description |
|---------------------------|---|
| TTA_NoPrintPrefs | Corresponds to the Windows application object's Printer Preference Caching attribute. |
| TTA_ParentName | The application object's fully qualified name. This variable provides the same information as TTA_ApplicationName. |
| TTA_PortNumber | Corresponds to the 3270 or 5250 application object's Server Port attribute. |
| TTA_ProtocolArguments | Corresponds to the Windows application object's Arguments attribute. |
| TTA_RemoteAddr | The IP address of the application server used to run the application. |
| TTA_RemoteAudio | Corresponds to the Windows application object's Remote Audio attribute. |
| TTA_RequiresDisplayEngine | Whether or not the application requires a display engine. |
| TTA_ResumeTimeOut | Corresponds to the application object's Application Resumability: Timeout attribute. |
| TTA_RootColor | Corresponds to the application object's Window Color: Custom Color attribute. |
| TTA_RootType | Corresponds to the application object's Window Color attribute. This variable can have the value default, or color. |
| TTA_ScrollStyle | Corresponds to the character application object's Scroll Style attribute. This variable can have the value normal, jump, or smooth. |
| TTA_SecureConnection | Corresponds to the user profile's Security tab. |
| TTA_SessionExit | Corresponds to the application object's Session Termination attribute. This variable can have the value lastclient - Last Client Exit, windowmanager - Window Manager Exit, windowmanageralone - Only Window Manager Remaining), loginscript - Login Script Exit), loginscript - Login Script Exit), nowindows - No Visible Windows, and loginscriptnowindows - Login Script exit or No Visible Windows. |
| TTA_SettingsItem | Corresponds to the 3270 or 5250 application object's 'File' and 'Settings' Menus attribute. |
| TTA_StatusLine | Corresponds to the character application object's Status Line attribute. This variable can have the value none, indicator, and host writable, standard, or extended. |
| TTA_Suspend | Corresponds to the application object's Application Resumability attribute. This variable can have the value never, session (means User Session), and forever (means Always). |

| Variable | Description |
|-------------------------------|--|
| TTA_TerminalClass | Corresponds to the character application object's Emulation Type attribute. This variable can have the value <code>scoconsole</code> , <code>vt420</code> , or <code>wyse60</code> . |
| TTA_TerminalType | Corresponds to the character application object's Terminal Type attribute. |
| TTA_TNClose | Corresponds to the 3270 or 5250 application object's Connection Closed Action attribute. This variable can have the value 0 - Prompt User for Action, 1 - Exit Emulator, 2 - Reconnect, and 3 - Close Connection. |
| TTA_TopMenuBar | Corresponds to the 3270 or 5250 application object's Menu Bar attribute. |
| TTA_Transport | Corresponds to the application object's Connection Method attribute. This variable can have the value <code>rexec</code> , <code>telnet</code> , or <code>ssh</code> . The guaranteed variable <code>TTA_CONNECTIONSERVICE</code> also provides this information. |
| TTA_UserName | The fully qualified name of the user this application session is for. |
| TTA_UserSecurityEquivalent | Corresponds to the user profile's User Name attribute. |
| TTA_UNIXAUDIO_QUALITY | Corresponds to the UNIX Audio Quality attribute on the Global Settings → Client Device tab in the Administration Console. This variable can have the value <code>low</code> , <code>medium</code> , and <code>high</code> . |
| TTA_UNIXAUDIOPRELOAD | Corresponds to the X application objects Audio Redirection Library attribute. |
| TTA_ViewHostReply | Corresponds to the application object's Keep Launch Connection Open attribute. |
| TTA_WebTop | Corresponds to the Webtop Theme attribute. Note - This attribute is no longer used. |
| TTA_Width | Corresponds to the application object's Window Size: Width attribute. This variable provides the same information as <code>TTA_AppletWidth</code> . |
| TTA_WinCursor | Corresponds to the application object's Use Windows Cursor attribute. Note - This attribute is no longer used. |
| TTA_WindowsApplicationServer | The protocol used to connect to a Windows application server. This variable can have the value <code>wincenter</code> , <code>wincentermf</code> - Citrix UNIX Integration Services, <code>merge</code> - SCO Merge, <code>winframe</code> - Citrix ICA), <code>wts</code> - Microsoft RDP, or <code>none</code> . Only Microsoft RDP is supported. The other protocols can only be used with legacy SGD Windows application objects. |
| TTA_WindowsApplicationSupport | Corresponds to the Windows application object's Local Client Launch attribute. |
| TTA_WorkingDir | Corresponds to the Windows application object's Working Directory attribute. |

Login Script Timeouts

SGD uses several timeouts when starting applications. The following timeouts are available:

- [“Expect Timeouts” on page 874](#)
- [“Client Timers” on page 875](#)
- [“Other Timeouts” on page 877](#)

Note – None of the timeouts, apart from the Execution Protocol Engine timeout, apply when starting a Microsoft Windows application.

Expect Timeouts

The Expect timeouts are defined in the `vars.exp` login script. The following table lists the available Expect timeouts and their default values.

| Timeout | Default Value |
|----------------------------------|---------------|
| <code>timeouts(hostprobe)</code> | 30 seconds |
| <code>timeouts(prelogin)</code> | 40 seconds |
| <code>timeouts(loggedin)</code> | 20 seconds |

If an Expect timeout expires, the script attempts to guess the prompt, and then continues to start the application.

`timeouts(hostprobe)`

The `timeouts(hostprobe)` timeout is called by the `unix.exp` login script. This is the time to wait for a response from the `ttahostprobe` binary. The `ttahostprobe` binary is used to check whether an application server is available.

The `ttahostprobe` binary outputs its response to standard output (`stdout`), and returns `y` for success or `n` for failure.

timeouts(prelogin)

The time allowed for each Expect command to match a required string during the login phase.

For example, after the connection is made to the application server, the script has 40 seconds by default to match the login prompt before it times out. Every successful match resets the timer. During a login, the timeout is usually reset for the login prompt, the password prompt, and the shell prompt.

Increasing this timeout increases the time allowed for each phase of the login. This timeout must be large enough to allow for the longest phase of the login to be completed.

If the timeout expires, the script assumes that it is logged in and has failed to match the shell prompt and sends “echo SYNC” to the application server to guess the prompt string. If the user is not logged in when the timer fires, the application fails to start. Otherwise, the shell prompt is set to whatever the application server sent immediately after the “echo SYNC” and the application startup continues.

Note – If you see “echo SYNC” and the shell prompt ends in the normal way with \$, %, #, or >, the `timeouts(prelogin)` value is too short.

timeouts(loggedin)

The time allowed for each Expect command to match a required string once the user is logged in.

If the timeout expires, the script moves on to the next command. This can cause commands to be sent before the prompt has returned.

The most common occurrence of this timeout is if the script incorrectly sets the shell prompt. By default, this causes each command to wait 20 seconds before moving to the next command and can trigger one of the client timers.

Client Timers

Client timers are set using the `clienttimer` Tcl command (see “[clienttimer](#)” on [page 862](#)). If a client timer expires, the application start is canceled with a fatal `ErrApplicationServerTimeout` error.

The client timers are defined in the `vars.exp` login script.

The following table lists the available client timers and their default values.

| Timer | Default Value |
|------------------------------|--|
| <code>timers(login)</code> | <code>timeouts(prelogin)</code> + 10 seconds |
| <code>timers(env)</code> | 40 seconds |
| <code>timers(runmain)</code> | 40 seconds |
| <code>timers(build)</code> | 25 seconds |
| <code>timers(total)</code> | 5 seconds |

`timers(login)`

The total time for the complete login phase, from making the connection to receiving the first shell prompt.

The `timers(login)` timer must be large enough to cover all of the login phases. Each individual phase of the login (login prompt, password prompt, shell prompt) might last up to the number of seconds defined for the `timeouts(prelogin)` timeout. The value of this timer must always be greater than `timeouts(prelogin)` Expect timeout.

If you increase the `timeouts(prelogin)` Expect timeout, increase the `timers(login)` timer as well so that the difference between them is never less than 10.

`timers(env)`

The total time from receiving the first shell prompt until all of the application server environment variables have been exported.

`timers(runmain)`

The total time from setting the last environment variable to starting the main application.

`timers(build)`

The total time taken to build the command line to be executed. This timer is only used when starting Windows applications that use the SCO Merge protocol.

Note – The SCO Merge protocol is no longer supported and can only be used by legacy SGD Windows application objects.

`timers(total)`

The total number of client timers. Only change this setting if you add or remove a client timer.

Other Timeouts

The `procs.exp` login script includes a three-second timeout when issuing commands. This is defined in the `proc wait_for_prompt` procedure.

The Execution Protocol Engine has a default timeout of 180 seconds (three minutes). This timeout starts when the request to start an application is received and removed when the application startup has completed successfully. If it expires, the application startup is canceled. This timeout is specific to each SGD server.

Use the following command to change this timeout:

```
$ tarantella config edit \  
--tarantella-config-execpeconfig-maxlaunchtime secs
```

Note – Use the `--array` option with this command to change this timeout for all the SGD servers in the array.

Login Script Error Messages

The following table lists the error codes and messages that can occur with login scripts, and a description of what to do about them. Use this information to diagnose why a login script is failing.

| Code | Error Message and Description |
|------|--|
| 0 | <code>ErrOK</code> The login script successfully connected to the application server and started the application. |
| 1 | <code>ErrApplicationServerResourceFailure</code> The login script failed due to a lack of system resources on the application server. Ensure that the application server is capable of running the application. |
| 2 | <code>ErrApplicationServerNoLicenseAvailable</code> No licenses were available on the application server. Ensure that the application server has sufficient licenses for the number of connections you expect to make. |
| 3 | <code>ErrFaultInExecutionScript</code> The login script contains a syntax error. Review the login script. |
| 4 | <code>ErrApplicationServerLoginFailed</code> The login script failed to log into the application server. See see “Troubleshooting ErrApplicationServerLoginFailed Errors” on page 206 . |
| 5 | <code>ErrApplicationServerLoginIncorrect</code> The user name and password supplied to the application server were not accepted. Check that the user name and password are valid on that application server. |
| 6 | <code>ErrApplicationServerPasswordAged</code> The user’s password on the application server has expired. Ensure that the user has a valid password on the application server. To avoid seeing this error, configure SGD to handle aged passwords. You configure this on the Global Settings → Application Authentication tab in the Administration Console. |
| 7 | <code>ErrCommandExecutionFailed</code> The login script successfully logged in to the application server but could not run the application. Ensure that the application object’s Application Command attribute contains a valid command. Ensure that the user has write permissions for the /tmp directory on the application server. |
| 8 | <code>ErrApplicationServerConnectionFailed</code> The login script failed to log in to the application server. Check that you can log into the application server manually. |

| Code | Error Message and Description |
|------|--|
| 9 | <p><code>ErrApplicationServerEndOfFileOnConnection</code></p> <p>The login script encountered an End of File error (EOF) on connection to the application server. Investigate why an EOF error is returned.</p> |
| 10 | <p><code>ErrApplicationServerTimeout</code></p> <p>The login script timed out when trying to connect to the application server. See “Troubleshooting ErrApplicationServerTimeout Errors” on page 206.</p> |
| 12 | <p><code>ErrInvalidDesktopSize</code></p> <p>The width and height defined for a Windows application is not valid. Check the application object’s Window Size: Width and Window Size: Height attributes.</p> |
| 14 | <p><code>ErrCouldNotPipe</code></p> <p>The login script was unable to create a pipe between the parent and child processes in the Execution Protocol Engine. This error might indicate that there is not enough memory on the application server. Check the number of other applications running on the server and increase size of memory if necessary.</p> |
| 15 | <p><code>ErrCouldNotFork</code></p> <p>The login script was unable to fork a child process in the Execution Protocol Engine. This error might indicate that there is not enough memory on the application server. Check the number of other applications running on the server and increase the amount of memory if necessary.</p> |
| 16 | <p><code>ErrScriptRead</code></p> <p>The login script produced an error when trying to read from the script process in the Execution Protocol Engine. Try to run the application again. If the error persists, contact Oracle Support.</p> |
| 17 | <p><code>ErrScriptWrite</code></p> <p>The login script produced an error when trying to write to the script process in the Execution Protocol Engine. Try to run the application again. If the error persists, contact Oracle Support.</p> |
| 18 | <p><code>ErrThirdTierWrite</code></p> <p>The login script produced an error when trying to write to the application server in the Execution Protocol Engine. This error usually means the connection to the application server was lost. Check the application server is available and try to run the application again.</p> |
| 19 | <p><code>ErrThirdTierRead</code></p> <p>The login script produced an error when trying to read from the application server in the Execution Protocol Engine. This error usually means the connection to the application server was lost. Check the application server is available and try to run the application again.</p> |

| Code | Error Message and Description |
|------|--|
| 21 | <p><code>ErrTransportNotAvailable</code></p> <p>The login script was unable to connect to the application server using the requested connection method method.</p> <p>Check that the application server supports the connection method. Check that the application server is available.</p> |
| 22 | <p><code>ErrLogFileError</code></p> <p>This is not an application startup error. SGD was unable to create a log file for the Protocol Engine Manager.</p> <p>If the error persists, contact Oracle Support.</p> |
| 27 | <p><code>ErrThirdTierFailure</code></p> <p>Something has gone wrong on the application server.</p> <p>Check that the server is available and that you can run the application manually.</p> |
| 30 | <p><code>ErrLoginPasswordNotAvailable</code></p> <p>The login script was unable to supply the application server with a password.</p> <p>This error usually means the Execution Protocol Engine timeout has been triggered. See “Other Timeouts” on page 877 for details of how to increase the Execution Protocol Engine timeout.</p> |
| 31 | <p><code>ErrRequestNotSupported</code></p> <p>The login script cannot execute the requested auxiliary commands.</p> <p>Check that the Arguments for Command attribute for the application object is configured correctly and that the additional commands work on the application server.</p> |
| 32 | <p><code>ErrRequestNotImplemented</code></p> <p>The login script cannot execute the requested operation because it has not been implemented.</p> <p>If the error persists, contact Oracle Support.</p> |
| 33 | <p><code>ErrUnknown</code></p> <p>An error occurred in the Execution Protocol Engine.</p> <p>Check the log file and try to run the application again.</p> |
| 34 | <p><code>ErrInternalError</code></p> <p>An error in the Protocol Engine Manager.</p> <p>Check the log file and try to run the application again.</p> |
| 37 | <p><code>ErrProtocolEngineDied</code></p> <p>The Protocol Engine process failed.</p> <p>Check the log file for the process ID of the protocol engine and try running the application again. If the problem persists, contact Oracle Support.</p> |
| 43 | <p><code>ErrExpectInitialisationFailed</code></p> <p>SGD was unable to initialize the Expect interpreter and so the script was not run.</p> <p>Try to run the application again. If the problem persists, contact Oracle Support.</p> |

| Code | Error Message and Description |
|------|--|
| 44 | <p><code>ErrEvalFileFailed</code></p> <p>The login script file does not exist or contains a syntax error that is causing the Expect interpreter to fail. Check that the login script is in the specified directory. All login scripts supplied by SGD are stored in the <code>/opt/tarantella/var/serverresources/expect</code> directory. Check the Execution Protocol Engine error log file for details of any errors with the script.</p> |
| 45 | <p><code>ErrCreateInterpreterFailed</code></p> <p>SGD was unable to initialize the Tcl interpreter and so the script was not run. Try to run the application again. If the error persists, contact Oracle Support.</p> |
| 46 | <p><code>ErrChdirFailed</code></p> <p>The login script failed to change to the directory containing the script. Check the path to the script.</p> |
| 47 | <p><code>ErrReadError</code></p> <p>The login script produced an error when trying reading from the protocol connection between the parent and child processes in the Execution Protocol Engine. Try to run the application again. If the error persists, contact Oracle Support.</p> |
| 49 | <p><code>ErrEndOfFile</code></p> <p>The login script read an unexpected end of file on a connection. Try to run the application again. If the error persists, contact Oracle Support.</p> |
| 51 | <p><code>ErrBadMessage</code></p> <p>The login script received an invalid message, probably due to a corruption of the data packet. Try to run the application again. If the error persists, contact Oracle Support.</p> |
| 52 | <p><code>ErrStaleCookie</code></p> <p>The client connected to the application but the cookie needed for the application startup has expired. Try to run the application again.</p> <p>If this fails, increase the lifetime of the cookie. You do this with the following command:</p> <pre>\$ tarantella config edit --tarantella-config-applaunch-reconnecttimeout <i>seconds</i></pre> <p>The default value is 60 (60 seconds). If the error persists, contact Oracle Support.</p> |
| 53 | <p><code>ErrEatenCookie</code></p> <p>The client connected to the application but the cookie needed for the application startup has already been used, probably by the user running multiple sessions. Try to run the application again. If the error persists, contact Oracle Support.</p> |
| 54 | <p><code>ErrDifferentCookie</code></p> <p>The client connected to the application but the cookie supplied does not match the one required for the application startup. Try to run the application again. If the error persists, contact Oracle Support.</p> |
| 55 | <p><code>ErrLaunchPolicyNotFound</code></p> <p>SGD was unable to find the details needed to run the application. This error might never occur. Try to run the application again. If this fails, stop the SGD server, start it again, and then run the application again. If the error persists, contact Oracle Support.</p> |

| Code | Error Message and Description |
|------|---|
| 56 | <p><code>ErrBadLength</code></p> <p>The login script received a message that was not the correct length, probably due to a corruption of the data packet.</p> <p>Try to run the application again. If the error persists, contact Oracle Support.</p> |
| 57 | <p><code>ErrInvalidConfigObject</code></p> <p>The configuration data provided by SGD did not contain all the required information.</p> <p>This error might never occur. Try to run the application again. If this fails, stop the SGD server, start it again and then run the application. If the error persists, contact Oracle Support.</p> |
| 58 | <p><code>ErrSessionCircuitNotFound</code></p> <p>The connection between the protocol engine and the Protocol Engine Manager was lost.</p> <p>Try to run the application again. If this fails, stop the SGD server, start it again and then run the application. If the error persists, contact Oracle Support.</p> |
| 59 | <p><code>ErrExecutionCircuitNotFound</code></p> <p>The connection between the Protocol Engine Manager and the Execution Protocol Engine was lost.</p> <p>Try to run the application again. If this fails, stop the SGD server, start it again, and then run the application. If the error persists, contact Oracle Support.</p> |
| 61 | <p><code>ErrCircuitNotFound</code></p> <p>The Protocol Engine Manager cannot find a circuit (connection).</p> <p>Try to run the application again. If this fails, stop the SGD server, start it again and then run the application. If the error persists, contact Oracle Support.</p> |
| 62 | <p><code>ErrCreateFailed</code></p> <p>The create request to the protocol engine failed and SGD was unable to run the application.</p> <p>The definition of the application is missing some attributes. Check the log file for details of the missing attributes and correct these errors before trying to run the application again</p> |
| 63 | <p><code>ErrComplete</code></p> <p>This is not an error. It is a message from the Execution Protocol Engine to the Protocol Engine Manager to indicate the launch process was completed.</p> |
| 65 | <p><code>ErrNonZeroConnectresult</code></p> <p>When the SGD Client connected to the protocol engine, an error occurred.</p> <p>If possible, log out. Otherwise, close the browser and end the SGD Client processes on the client device.</p> <p>Try to run the application again.</p> |
| 66 | <p><code>ErrUserAbort</code></p> <p>This is not an error. The user canceled the application launch.</p> |
| 67 | <p><code>ErrClientEndOfFileOnConnection</code></p> <p>The connection to the SGD Client was lost.</p> <p>If possible, log out. Otherwise, close the browser and end the SGD Client processes on the client device.</p> <p>Try to run the application again.</p> |

| Code | Error Message and Description |
|------|---|
| 68 | <p data-bbox="215 192 419 227"><code>ErrNothingToDo</code></p> <p data-bbox="215 227 429 262">This is not an error.</p> <p data-bbox="215 262 1316 322">This message indicates that the launch request sent to the Protocol Engine Manager does not require any protocol engines.</p> |
| 71 | <p data-bbox="215 331 362 366"><code>ErrIoError</code></p> <p data-bbox="215 366 722 401">The login script was unable to write to <code>stderr</code>.</p> <p data-bbox="215 401 1019 435">Try to run the application again. If the error persists, contact Oracle Support.</p> |
| 73 | <p data-bbox="215 444 472 479"><code>ErrTscLicenseError</code></p> <p data-bbox="215 479 1268 513">There are not enough Windows Terminal Services licenses available to be able to run the application.</p> <p data-bbox="215 513 848 543">Increase the number of Windows Terminal Services licenses.</p> |

Third-Party Legal Notices

This appendix contains the legal notices for third-party software used by Oracle Secure Global Desktop (SGD). The following licenses are included:

- [“Apache License, Version 2.0” on page 886](#)
- [“Cryptix” on page 892](#)
- [“JACL/TCL” on page 892](#)
- [“OpenMotif” on page 894](#)
- [“OpenSSL” on page 899](#)
- [“TeemTalk” on page 902](#)
- [“Unicode Character Database” on page 902](#)
- [“X Window System \(X11R6.8.2\)” on page 910](#)
- [“XML Parser” on page 910](#)
- [“zlib” on page 911](#)

Apache License, Version 2.0

Apache HTTP Server
Copyright 2009 The Apache Software Foundation.

This product includes software developed at The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were developed at the National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign.

This software contains code derived from the RSA Data Security Inc. MD5 Message-Digest Algorithm, including various modifications by Spyglass Inc., Carnegie Mellon University, and Bell Communications Research, Inc (Bellcore).

Regular expression support is provided by the PCRE library package, which is open source software, written by Philip Hazel, and copyright by the University of Cambridge, England. The original software is available from

<ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcre/>

Apache Tomcat
Copyright 1999-2010 The Apache Software Foundation

This product includes software developed by The Apache Software Foundation (<http://www.apache.org/>).

The Windows Installer is built with the Nullsoft Scriptable Install System (NSIS), which is open source software. The original software and related information is available at <http://nsis.sourceforge.net>.

Java compilation software for JSP pages is provided by Eclipse, which is open source software. The original software and related information is available at <http://www.eclipse.org>.

```
=====
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Axis distribution.      ==
=====
```

This product includes software developed by The Apache Software Foundation (<http://www.apache.org/>).

The following applies to all products licensed under the Apache 2.0 License: You may not use the identified files except in compliance with the Apache License, Version 2.0 (the "License.")

You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0>. A copy of the license is also reproduced below.

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

License: Apache 2.0, 2004; <http://www.apache.org/licenses/LICENSE-2.0>

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You

meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the

origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache Projects

Copyright © 2010 The Apache Software Foundation, Licensed under the Apache
License, Version 2.0.

Apache and the Apache feather logo are trademarks of The Apache Software
Foundation.

Cryptix

Cryptix General License

Copyright (c) 1995-2005 The Cryptix Foundation Limited.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE CRYPTIX FOUNDATION LIMITED AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE CRYPTIX FOUNDATION LIMITED OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

JACL/TCL

The following terms apply to all versions of the core Tcl/Tk releases, the Tcl/Tk browser plug-in version 2.0, and TclBlend and Jacl version 1.0. Please note that the TclPro tools are under a different license agreement. This agreement is part of the standard Tcl/Tk distribution as the file named "license.terms".

TCL/TK LICENSE TERMS

This software is copyrighted by the Regents of the University of California, Sun Microsystems, Inc., Scriptics Corporation, and other parties. The following terms apply to all files associated with the software unless explicitly disclaimed in individual files.

The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

GOVERNMENT USE: If you are acquiring this software on behalf of the U.S. government, the Government shall have only "Restricted Rights" in the software and related documentation as defined in the Federal Acquisition Regulations (FARs) in Clause 52.227.19 (c) (2). If you are acquiring the software on behalf of the Department of Defense, the software shall be classified as "Commercial Computer Software" and the Government shall have only "Restricted Rights" as defined in Clause 252.227-7013 (c) (1) of DFARs. Notwithstanding the foregoing, the authors grant the U.S. Government and others acting in its behalf permission to use and distribute the software in accordance with the terms specified in this license.

OpenMotif

THE OPEN GROUP PUBLIC LICENSE

MOTIF GRAPHICAL USER INTERFACE SOFTWARE

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS THE OPEN GROUP PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

1. in the case of The Open Group, L.L.C. ("The Open Group"), the Original Program, and
2. in the case of each Contributor,
 - i. changes to the Program, and
 - ii. additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which:

- i. are separate modules of software distributed in conjunction with the Program under their own license agreement, even if the separate modules are linked in binary form to the Program, and
- ii. are not derivative works of the Program.

"Contributor" means The Open Group and any other entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Open Source" programs mean software for which the source code is available without confidential or trade secret restrictions and for which the source code and object code are available for distribution without license charges.

"Original Program" means the original version of the software accompanying this

Agreement as released by The Open Group, including source code, object code and documentation, if any.

"Program" means the Original Program and Contributions.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

The rights granted under this license are limited solely to distribution and sublicensing of the Contribution(s) on, with, or for operating systems which are themselves Open Source programs. Contact The Open Group for a license allowing distribution and sublicensing of the Original Program on, with, or for operating systems which are not Open Source programs.

1. Subject to the terms of this Agreement and the limitations of this Section 2, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

Section 2, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

3. Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

4. Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its

own license agreement, provided that:

- a. it complies with the terms and conditions of this Agreement; and
- b. its license agreement:

fitness for a particular purpose; non-infringement, and implied warranties or conditions of merchantability and conditions, express and implied, including warranties or conditions of title and i. effectively disclaims on behalf of all Contributors all warranties and lost profits; including direct, indirect, special, incidental and consequential damages, such as ii. effectively excludes on behalf of all Contributors all liability for damages, that Contributor alone and not by any other party; and iii. states that any provisions which differ from this Agreement are offered by code form: customarily used for software exchange. When the Program is made available in source informs licensees how to obtain it in a reasonable manner on or through a medium iv. states that source code for the Program is available from such Contributor, and

- a. it must be made available under this Agreement; and
- b. a copy of this Agreement must be included with each copy of the Program.

Each Contributor must include the following in a conspicuous location in the Program:

Copyright (c) {date here}, The Open Group Ltd. and others. All Rights Reserved.

In addition, each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, subject to the limitations provided in Section 2, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

- a. promptly notify the Commercial Contributor in writing of such claim, and
- b. allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defence and any related settlement negotiations.

The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation or other similar official proceedings to enforce patent rights against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit),

then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such non-compliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

The Open Group may publish new versions (including revisions) of this Agreement from time to time. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. No one other than The Open Group has the right to modify this Agreement. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

OpenSSL

License

This is a copy of the current LICENSE file inside the CVS repository.

LICENSE ISSUES
=====

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License

```
/* =====  
 * Copyright (c) 1998-2008 The OpenSSL Project. All rights reserved.  
 *  
 * Redistribution and use in source and binary forms, with or without  
 * modification, are permitted provided that the following conditions  
 * are met:  
 *  
 * 1. Redistributions of source code must retain the above copyright  
 * notice, this list of conditions and the following disclaimer.  
 *  
 * 2. Redistributions in binary form must reproduce the above copyright  
 * notice, this list of conditions and the following disclaimer in  
 * the documentation and/or other materials provided with the  
 * distribution.  
 *  
 * 3. All advertising materials mentioning features or use of this  
 * software must display the following acknowledgment:  
 * "This product includes software developed by the OpenSSL Project  
 * for use in the OpenSSL Toolkit. (http://www.openssl.org/)"  
 *  
 * 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to  
 * endorse or promote products derived from this software without  
 * prior written permission. For written permission, please contact  
 * openssl-core@openssl.org.  
 *  
 * 5. Products derived from this software may not be called "OpenSSL"  
 * nor may "OpenSSL" appear in their names without prior written  
 * permission of the OpenSSL Project.
```

```

*
* 6. Redistributions of any form whatsoever must retain the following
*      acknowledgment:
*      "This product includes software developed by the OpenSSL Project
*      for use in the OpenSSL Toolkit (http://www.openssl.org/)"
*
* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY
* EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
* PURPOSE ARE DISCLAIMED.  IN NO EVENT SHALL THE OpenSSL PROJECT OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
* OF THE POSSIBILITY OF SUCH DAMAGE.
* =====
*
* This product includes cryptographic software written by Eric Young
* (eay@cryptsoft.com).  This product includes software written by Tim
* Hudson (tjh@cryptsoft.com).
*
*/

Original SSLeay License
-----

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are aheared to.  The following conditions
 * apply to all code found in this distribution, be it the RC4, RSA,
 * lhash, DES, etc., code; not just the SSL code.  The SSL documentation
 * included with this distribution is covered by the same copyright terms
 * except that the holder is Tim Hudson (tjh@cryptsoft.com).
 *
 * Copyright remains Eric Young's, and as such any Copyright notices in
 * the code are not to be removed.
 * If this package is used in a product, Eric Young should be given attribution
 * as the author of the parts of the library used.
 * This can be in the form of a textual message at program startup or
 * in documentation (online or textual) provided with the package.

```

```

*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. All advertising materials mentioning features or use of this software
*    must display the following acknowledgement:
*    "This product includes cryptographic software written by
*    Eric Young (eay@cryptsoft.com)"
*    The word 'cryptographic' can be left out if the routines from the library
*    being used are not cryptographic related :-).
* 4. If you include any Windows specific code (or a derivative thereof) from
*    the apps directory (application code) you must include an acknowledgement:
*    "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
*
* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*/

```

TeemTalk

Some portions of this computer software product are copyright works owned by Neoware UK Ltd.

Copyright (c) Neoware UK Ltd. 1990-2005

Oracle and/or its affiliates is acting as a distributor and not as author or developer of this software.

Unicode Character Database

UNICODE 2.1 CHARACTER DATABASE

Copyright (c) 1991-1998 Unicode, Inc.
All Rights reserved.

DISCLAIMER

The Unicode Character Database "UNIDAT21.TXT" is provided as-is by Unicode, Inc. (The Unicode Consortium). No claims are made as to fitness for any particular purpose. No warranties of any kind are expressed or implied. The recipient agrees to determine applicability of information provided. If this file has been purchased on magnetic or optical media from Unicode, Inc., the sole remedy for any claim will be exchange of defective media within 90 days of receipt.

This disclaimer is applicable for all other data files accompanying the Unicode Character Database, some of which have been compiled by the Unicode Consortium, and some of which have been supplied by other vendors.

LIMITATIONS ON RIGHTS TO REDISTRIBUTE THIS DATA

Recipient is granted the right to make copies in any form for internal distribution and to freely use the information supplied in the creation of products supporting the Unicode (TM) Standard. This file can be redistributed to third parties or other organizations (whether for profit or not) as long as this notice and the disclaimer notice are retained.

EXPLANATORY INFORMATION

The Unicode Character Database defines the default Unicode character properties, and internal mappings. Particular implementations may choose to override the properties and mappings that are not normative. If that is done, it is up to the implementer to establish a protocol to convey that information. For more information about character properties and mappings, see "The Unicode Standard, Worldwide Character Encoding, Version 2.0", published by Addison-Wesley. For information about other data files accompanying the Unicode Character Database, see the section of the Unicode Standard they were extracted from, or the explanatory readme files and/or header sections with those files.

The Unicode Character Database has been updated to reflect Version 2.1 of the Unicode Standard, with two additional characters added to those published in Version 2.0:

U+20AC EURO SIGN
U+FFFC OBJECT REPLACEMENT CHARACTER

A number of corrections have also been made to case mappings or other errors in the database noted since the publication of Version 2.0. And a few normative bidirectional properties have been modified to reflect decisions of the Unicode Technical Committee.

The Unicode Character Database is a plain ASCII text file consisting of lines containing fields terminated by semicolons. Each line represents the data for one encoded character in the Unicode Standard, Version 2.1. Every encoded character has a data entry, with the exception of certain special ranges, as detailed below.

There are five special ranges of characters that are represented only by their start and end characters, since the properties in the file are uniform, except for code values (which are all sequential and assigned). The names of CJK ideograph characters and Hangul syllable characters are algorithmically derivable. (See the Unicode Standard for more information). Surrogate characters and private use characters have no names.

The exact ranges represented by start and end characters are:

The CJK Ideographs Area (U+4E00 - U+9FFF)
The Hangul Syllables Area (U+AC00 - U+D7A3)
The Surrogates Area (U+D800 - U+DFFF)
The Private Use Area (U+E000 - U+F8FF)
CJK Compatibility Ideographs (U+F900 - U+FAFF)

The following table describes the format and meaning of each field in a data entry in the Unicode Character Database. Fields which contain normative information are so indicated.

| Field | Explanation |
|-------|-------------|
|-------|-------------|

-
- 0 Code value in 4-digit hexadecimal format.
This field is normative.
 - 1 Unicode 2.1 Character Name. These names match exactly the names published in Chapter 7 of the Unicode Standard, Version 2.0, except for the two additional characters.
This field is normative.
 - 2 General Category. This is a useful breakdown into various "character types" which can be used as a default categorization in implementations. Some of the values are normative, and some are informative. See below for a brief explanation.
 - 3 Canonical Combining Classes. The classes used for the Canonical Ordering Algorithm in the Unicode Standard. These classes are also printed in Chapter 4 of the Unicode Standard. This field is normative. See below for a brief explanation.
 - 4 Bidirectional Category. See the list below for an explanation of the abbreviations used in this field. These are the categories required by the Bidirectional Behavior Algorithm in the Unicode Standard. These categories are summarized in Chapter 4 of the Unicode Standard. This field is normative.
 - 5 Character Decomposition. In the Unicode Standard, not all of the decompositions are full decompositions. Recursive application of look-up for decompositions will, in all cases, lead to a maximal decomposition. The decompositions match exactly the decompositions published with the character names in Chapter 7 of the Unicode Standard. This field is normative.
 - 6 Decimal digit value. This is a numeric field. If the character has the decimal digit property, as specified in Chapter 4 of the Unicode Standard, the value of that digit is represented with an integer value in this field. This field is normative.
 - 7 Digit value. This is a numeric field. If the character represents a digit, not necessarily a decimal digit, the value is here. This covers digits which do not form decimal radix forms, such as the compatibility superscript digits. This field is informative.
 - 8 Numeric value. This is a numeric field. If the character has the numeric property, as specified in Chapter 4 of the Unicode Standard, the value of that character is represented with an integer or rational number in this field. This includes fractions as, e.g., "1/5" for U+2155 VULGAR FRACTION ONE FIFTH.
Also included are numerical values for compatibility characters

such as circled numbers. This field is normative.

- 9 If the character has been identified as a "mirrored" character in bidirectional text, this field has the value "Y"; otherwise "N". The list of mirrored characters is also printed in Chapter 4 of the Unicode Standard. This field is normative.
- 10 Unicode 1.0 Name. This is the old name as published in Unicode 1.0. This name is only provided when it is significantly different from the Unicode 2.1 name for the character. This field is informative.
- 11 10646 Comment field. This field is informative.
- 12 Upper case equivalent mapping. If a character is part of an alphabet with case distinctions, and has an upper case equivalent, then the upper case equivalent is in this field. See the explanation below on case distinctions. These mappings are always one-to-one, not one-to-many or many-to-one. This field is informative.
- 13 Lower case equivalent mapping. Similar to 12. This field is informative.
- 14 Title case equivalent mapping. Similar to 12. This field is informative.

GENERAL CATEGORY

The values in this field are abbreviations for the following. Some of the values are normative, and some are informative. For more information, see the Unicode Standard. Note: the standard does not assign information to control characters (except for TAB in the Bidirectional Algorithm). Implementations will generally also assign categories to certain control characters, notably CR and LF, according to platform conventions.

Normative

Mn = Mark, Non-Spacing
Mc = Mark, Spacing Combining
Me = Mark, Enclosing

Nd = Number, Decimal Digit
Nl = Number, Letter
No = Number, Other

Zs = Separator, Space
Zl = Separator, Line
Zp = Separator, Paragraph

Cc = Other, Control
Cf = Other, Format
Cs = Other, Surrogate

Co = Other, Private Use
Cn = Other, Not Assigned

Informative

Lu = Letter, Uppercase
Ll = Letter, Lowercase
Lt = Letter, Titlecase
Lm = Letter, Modifier
Lo = Letter, Other

Pc = Punctuation, Connector
Pd = Punctuation, Dash
Ps = Punctuation, Open
Pe = Punctuation, Close
Po = Punctuation, Other

Sm = Symbol, Math
Sc = Symbol, Currency
Sk = Symbol, Modifier
So = Symbol, Other

BIDIRECTIONAL PROPERTIES

Please refer to the Unicode Standard for an explanation of the algorithm for Bidirectional Behavior and an explanation of the significance of these categories. These values are normative.

Strong types:

L Left-Right; Most alphabetic, syllabic, and logographic characters (e.g., CJK ideographs)
R Right-Left; Arabic, Hebrew, and punctuation specific to those scripts

Weak types:

EN European Number
ES European Number Separator
ET European Number Terminator
AN Arabic Number
CS Common Number Separator

Separators:

B Block Separator
S Segment Separator

Neutrals:

WS Whitespace
ON Other Neutrals ; All other characters: punctuation, symbols

CHARACTER DECOMPOSITION TAGS

The decomposition is a normative property of a character. The tags supplied with certain decompositions generally indicate formatting information. Where no such tag is given, the decomposition is designated as canonical. Conversely, the presence of a formatting tag also indicates that the decomposition is a compatibility decomposition and not a canonical decomposition. In the absence of other formatting information in a compatibility decomposition, the tag is used to distinguish it from canonical decompositions.

In some instances a canonical decomposition or a compatibility decomposition may consist of a single character. For a canonical decomposition, this indicates that the character is a canonical equivalent of another single character. For a compatibility decomposition, this indicates that the character is a compatibility equivalent of another single character.

The compatibility formatting tags used are:

- A font variant (e.g. a blackletter form).
- A no-break version of a space or hyphen.
- An initial presentation form (Arabic).
- A medial presentation form (Arabic).
- A final presentation form (Arabic).
- An isolated presentation form (Arabic).
- An encircled form.
- A superscript form.
- A subscript form.
- A vertical layout presentation form.
- A wide (or zenkaku) compatibility character.
- A narrow (or hankaku) compatibility character.
- A small variant form (CNS compatibility).
- A CJK squared font variant.
- A vulgar fraction form.
- Otherwise unspecified compatibility character.

CANONICAL COMBINING CLASSES

- 0: Spacing, enclosing, reordrant, and surrounding
- 1: Overlays and interior
- 6: Tibetan subjoined Letters
- 7: Nuktas
- 8: Hiragana/Katakana voiced marks
- 9: Viramas
- 10: Start of fixed position classes
- 199: End of fixed position classes
- 200: Below left attached
- 202: Below attached
- 204: Below right attached
- 208: Left attached (reordrant around single base character)
- 210: Right attached

212: Above left attached
214: Above attached
216: Above right attached
218: Below left
220: Below
222: Below right
224: Left (reordrant around single base character)
226: Right
228: Above left
230: Above
232: Above right
234: Double above

Note: some of the combining classes in this list do not currently have members but are specified here for completeness.

CASE MAPPINGS

In addition to uppercase and lowercase, because of the inclusion of certain composite characters for compatibility, such as "01F1;LATIN CAPITAL LETTER DZ", there is a third case, called titlecase, which is used where the first character of a word is to be capitalized (e.g. UPPERCASE, Titlecase, lowercase). An example of such a character is "01F2;LATIN CAPITAL LETTER D WITH SMALL LETTER Z".

The uppercase, titlecase and lowercase fields are only included for characters that have a single corresponding character of that type. Composite characters (such as "339D;SQUARE CM") that do not have a single corresponding character of that type can be cased by decomposition.

The case mapping is an informative, default mapping. Certain languages, such as Turkish, German, French, or Greek may have small deviations from the default mappings listed in the Unicode Character Database.

MODIFICATION HISTORY

Modifications made in updating the Unicode Character Database for the Unicode Standard, Version 2.1 (from Version 2.0) are:

- * Added two characters (U+20AC and U+FFFC).
- * Amended bidi properties for U+0026, U+002E, U+0040, U+2007.
- * Corrected case mappings for U+018E, U+019F, U+01DD, U+0258, U+0275, U+03C2, U+1E9B.
- * Changed combining order class for U+0F71.
- * Corrected canonical decompositions for U+0F73, U+1FBE.
- * Changed decomposition for U+FB1F from compatibility to canonical.
- * Added compatibility decompositions for U+FBE8, U+FBE9, U+FBF9..U+FBFB.
- * Corrected compatibility decompositions for U+2469, U+246A, U+3358.

Some of the modifications made in updating the Unicode Character Database for the Unicode Standard, Version 2.0 are:

- * Fixed decompositions with TONOS to use correct NSM: 030D.

- * Removed old Hangul Syllables; mapping to new characters are in a separate table.

- * Marked compability decompositions with additional tags.

- * Changed old tag names for clarity.

decomposition.* Revision of decompositions to use first-level decomposition, instead of maximal

- * Correction of all known errors in decompositions from earlier versions.

- * Added control code names (as old Unicode names).

- * Added Hangul Jamo decompositions.

- * Added Number category to match properties list in book.

- * Fixed categories of Koranic Arabic marks.

- * Fixed categories of precomposed characters to match decomposition where possible.

- * Added Hebrew cantillation marks and the Tibetan script.

Area.* Added place holders for ranges such as CJK Ideographic Area and the Private Use

- * Added categories Me, Sk, Pc, Nl, Cs, Cf, and rectified a number of mistakes in the database.

X Window System (X11R6.8.2)

<http://xorg.freedesktop.org/releases/X11R6.8.2/doc/LICENSE3.html#3>

XFree86 License:

XFree86 code without an explicit copyright is covered by the following copyright/license:

Copyright (C) 1994-2003 The XFree86 Project, Inc. All Rights Reserved.
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE XFREE86 PROJECT BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of the XFree86 Project shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from the XFree86 Project.

Portions also covered by other licenses as noted in the above url.

XML Parser

Copyright (c) 1997, 1998 James Clark

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the ``Software''), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included

in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED ``AS IS'', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL JAMES CLARK BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of James Clark shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from James Clark.

zlib

License

```
/* zlib.h -- interface of the 'zlib' general purpose compression library
   version 1.2.2, October 3rd, 2004
```

Copyright (C) 1995-2004 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly jloup@gzip.org
Mark Adler madler@alumni.caltech.edu

*/

Glossary

Numeric

- 3270 Application object** An SGD object that represents a 3270 protocol application running on a mainframe host. 3270 Application objects have a `cn=` naming attribute.
- 5250 Application object** An SGD object that represents a 5250 protocol application running on an AS/400 host. 5250 Application objects have a `cn=` naming attribute.

A

- Active Directory** Microsoft's implementation of [LDAP directory services](#). Used to store information about the resources, services, and users across a [Windows domain](#).
- Active Directory Container object** An SGD object used to represent an [Active Directory](#) structure within the SGD organizational hierarchy. Active Directory Container objects have a `cn=` naming attribute.
- advanced load balancing** Load balancing algorithms that measure the true load on application servers, using information provided by the SGD [Enhancement Module](#).
- AIP** Adaptive Internet Protocol. A proprietary protocol used by SGD software components. AIP optimizes the user experience by choosing the most efficient ways to transfer application display data and user input between client devices and SGD servers.
- ALSA** Advanced Linux Sound Architecture.

| | |
|---|--|
| ambiguous login | The situation where an authentication mechanism has found more than one match for a user and cannot distinguish between them without further information from the user. |
| anonymous user authentication | An authentication mechanism where users can log in to SGD without supplying a user name or password. Anonymous user authentication is disabled by default. |
| ANSI | American National Standards Institute. |
| API | Application programming interface. |
| applet | A software program running in a web browser. |
| application launch dialog | Dialog shown when a user clicks a webtop link to start an application. |
| application load balancing | The mechanism that determines which application server runs a user's application. |
| application server | A networked device, such as a Windows 2000 server or Linux server, configured to run applications. Application servers are represented in the SGD datastore by an Application Server object . |
| Application Server object | An SGD object that represents an application server used to run applications through SGD. Application Server objects have a <code>cn=</code> naming attribute. |
| application server password cache | A secure store of application server user names and passwords associated with user identities. Maintained so that application server authentication can proceed without prompting the user. Also called the password cache. |
| application session | An application session begins when a user starts an application, and ends when the application exits. Information about an application session is stored in memory by the SGD server. Each application session is associated with a Protocol Engine . |
| application session load balancing | The mechanism that determines which SGD server in the array manages the application session, and runs the Protocol Engine for a user's application. |
| array | A collection of SGD servers that share configuration information. The SGD servers in an array act together to enable users to see the same webtop , and resume their applications, whatever SGD server they log in to. Arrays of SGD servers provide scalability and redundancy. |
| array route | Configures SOCKS proxy server usage, depending on the IP address of the client device. |
| Assignment Type | A field in the Administration Console that indicates the origin of an object link. Assignment Types can be Direct, Indirect, or Multiple. See also direct assignment , indirect assignment , multiple assignment . |
| ATR string | Automatic Terminal Recognition string. A sequence of bytes used to identify a smart card . |

| | |
|-----------------------------|---|
| attribute | A named property of an object. Attributes may have zero or more values, as defined by the schema. |
| attribute map | A file that defines how character attributes, such as bold and underline, are displayed in the SGD terminal emulators . |
| authentication token | In Integrated mode operation, identification data submitted from the SGD Client to the SGD server. Used by the authentication token authentication mechanism. |

B

| | |
|------------------------|---|
| batch scripting | The ability to perform more than one SGD related task with a single instance of a tarantella command . |
| billing service | An SGD service that logs user session and application session information for an SGD server or an array of SGD servers. |

C

| | |
|-------------------------------------|---|
| CA | See Certificate Authority . |
| CA certificate | See root certificate . |
| CAL | Client Access License. Used by Microsoft Windows Terminal Services . |
| CDE | Common Desktop Environment. A graphical user interface for UNIX desktops. |
| CDM | See client drive mapping . |
| Certificate Authority | A trusted issuer of SSL certificates . |
| Certificate Signing Request | Information supplied to a Certificate Authority , that is used to verify identity and generate an SSL certificate . |
| CGI | Common Gateway Interface. A specification for interfacing external applications with a web server. |
| Character Application object | An SGD object that represents a VT420, Wyse 60, or SCO Console application. Character Application objects have a <code>cn=</code> naming attribute. |
| cipher | In cryptography, an algorithm for performing encryption and decryption. |

| | |
|-----------------------------|--|
| client device | A networked device, such as a Windows PC or Linux workstation, used to access an SGD server. |
| client drive mapping | Enables users to access some or all of their client's drives, from an application running on an application server. |
| client profile | Settings for the SGD Client, including server URL, proxy settings, and mode of operation. The client profile is downloaded to the client device when a user connects to an SGD server. |
| CN | See common name . |
| color map | SGD terminal emulators support a palette of 16 colors. The color map is a file that defines the RGB values of these colors. |
| common name | A name used to identify an entry in an LDAP directory. For example, the name of a person. |
| COM port | A serial port , in a Microsoft Windows environment. |
| Configuration Wizard | A tool for SGD Administrators, useful for quickly adding new objects to an existing hierarchy, rather than creating a new hierarchy. |
| cookie | A short packet of data, used as an identification token. Some cookies are encrypted, to prevent forgery. |
| CPU | Central processing unit. |
| CSR | See Certificate Signing Request . |
| CUPS | Common UNIX Printing System. |

D

| | |
|-------------------------|--|
| daemon | A service process on UNIX platform operating systems that runs in the background, rather than under the direct control of a user. |
| data replication | The process where SGD system data is copied from the primary server in an SGD array to the secondary servers in the SGD array. |
| datastore | The sum of all the information used by the various components of SGD, including information about application servers and users on the network, user session and application session information, and organizational information. Organized into namespaces, such as <code>_ens</code> and <code>_dns</code> . |
| DER | Definite Encoding Rules. A cryptographic format used for storing SSL certificate keys. |
| DES | Data Encryption Standard. A cryptographic cipher . |

| | |
|---------------------------------------|---|
| digital signature | Information encrypted with a user's private key and appended to a message to ensure the authenticity of the message. The digital signature can be verified using the user's public key. See also public key cryptography . |
| direct assignment | In the Administration Console, a one-to-one object link created using the Editable Assignments table. See also editable assignment . |
| Directory (light) object | A container object in SGD, similar to an Organization object, but does not include SGD-specific attributes or allow you to assign applications. Examples include a Domain Component object and an Active Directory Container object. |
| directory services | Services that store and manage the resources and users on a network. SGD uses the principles of directory services for object storage and management. |
| Directory Services Integration | The ability to define webtops for users without requiring User Profile objects for those users in the SGD datastore. Instead, user information is kept in an external LDAP directory . Application objects in the SGD datastore define which LDAP users can see them on their webtop. |
| disambiguate | The process of resolving an ambiguous login. |
| Display Engine | An SGD software component that runs on a client device. Display Engines display applications to users and accept user input. They use AIP to communicate with Protocol Engines on SGD servers. |
| distinguished name | The name that uniquely identifies an entry in an LDAP directory . |
| distributed printing | Where print jobs are distributed across the array, avoiding bottlenecks and single points of failure. A user's print jobs are processed on the SGD server hosting the application session for the application you want to print from. |
| DN | See distinguished name . |
| DNS | Domain Name System. |
| DNS name | A unique name for a computer on a network, for example, <code>server.example.com</code> . |
| Document object | An SGD object that represents a document on the web. Documents can be any URL, including Sun StarOffice documents, or Adobe Acrobat files. A Document object can also refer to a web application. Document objects have a <code>cn=</code> naming attribute. |
| Domain Component object | An SGD object used to replicate a directory structure, usually a Microsoft Active Directory structure, within the SGD organizational hierarchy. Domain Component objects have a <code>dc=</code> naming attribute. |
| domain controller | See Windows domain controller . |
| DSI | See Directory Services Integration . |

E

| | |
|------------------------------|--|
| editable assignment | In the Administration Console, a one-to-one object link that can be edited by an SGD Administrator. See also direct assignment . |
| effective assignments | In the Administration Console, a summary of the object links for the current object. Effective assignments can include both direct assignments and indirect assignments . |
| Enhancement Module | An optional SGD software component installed on an application server to provide additional SGD functionality, such as client drive mapping , audio, and advanced load balancing . |
| environment variables | A set of system configuration values that can be accessed by a running program. |
| ESD | Enlightened Sound Daemon. A sound server for UNIX and Linux platforms that enables mixing of several digitized audio streams for playback by a single device. |
| EsoundD | See ESD . |
| ExecPE | Execution Protocol Engine. |
| Expect | An extension to the Tcl scripting language, typically used for interactive applications. The SGD login scripts are written in the Expect language. |
| external DNS name | The name by which an SGD server is known to a client device. An SGD server can have multiple external DNS names. |

F

| | |
|---------------------------|---|
| fingerprint | A short sequence of bytes used to authenticate or look up a public key . |
| FIPS | Federal Information Processing Standards. Standards developed by the United States Federal government for use by non-military government agencies and government contractors. |
| firewall traversal | Running SGD through a single open firewall port between client devices and SGD servers. Also known as firewall forwarding. |
| font server | A program that makes fonts on a host available on a network. |
| FQDN | See fully qualified domain name . |

| | |
|------------------------------------|--|
| fully qualified domain name | The full name of a system, containing its host name and its domain name. For example, <code>boston.example.com</code> , where <code>boston</code> is the host name of a server, and <code>example.com</code> is the domain name. |
| fully qualified name | An unambiguous name used to specify an SGD object. For example, <code>.../_ens/o=organization/ou=marketing/cn=Indigo Jones</code> , specifies a User Profile object in SGD. |

G

| | |
|------------------------------|---|
| Global Administrators | A role object in the Tarantella System Objects organization, used to assign administrative privileges to users. |
| global catalog | A domain controller that contains attributes for every object in the Active Directory . |
| Group object | An SGD object that represents a collection of applications or application servers. Each application or application server in the group is called a member . Group objects have a <code>cn=</code> naming attribute. |

H

| | |
|--------------|--|
| HTML | Hypertext Markup Language. A document format used for web pages. |
| HTTP | Hypertext Transfer Protocol. |
| HTTPS | Hypertext Transfer Protocol over Secure Sockets Layer. |

I

| | |
|-------------|---|
| IANA | Internet Assigned Numbers Authority. Organization that allocates and manages IP addresses, domain names, and port numbers used by the Internet. |
| ICA | Independent Computing Architecture. A protocol used by Citrix Presentation Server to communicate with client devices. |
| IM | See input method . |
| IME | Input method editor. See input method . |

| | |
|----------------------------|---|
| indirect assignment | In the Administration Console, an object link created by an LDAP search or by inheritance from another object. |
| inheritance | The ability to define webtop content implicitly. Content is usually inherited from the parent object, but other objects can also be used. |
| input method | A program that enable users to type in characters or symbols not found on their keyboard. On Microsoft Windows platforms, an IM is called an input method editor (IME). |
| Integrated mode | The mode of operation of SGD where your applications are displayed in the desktop Start or Launch menu. |
| I/O | Input/Output. |
| IP address | Internet Protocol address. A unique 32-bit numeric identifier for a computer on a network. |

J

| | |
|----------------------|--|
| JAR | Java Archive. |
| JDK | Java Development Kit. |
| JDS | Java Desktop System. |
| JRE | Java Runtime Environment. |
| JSP | JavaServer Page. |
| JSP container | A web server component that handles requests for JSP pages. SGD uses the Tomcat JSP container. |
| JSSE | Java Secure Socket Extension. An implementation of SSL using Java technology. |
| JVM | Java Virtual Machine. |

K

| | |
|------------|--|
| KDC | Key Distribution Center. Used by Kerberos authentication as part of the Active Directory authentication mechanism. |
| KDE | K Desktop Environment. An open source graphical user interface for UNIX and Linux platforms. |

| | |
|---------------------|--|
| Kerberos | An authentication system used for Active Directory authentication. |
| keyboard map | A file that contains mapping information between keys on the user's client keyboard and keys on a terminal. Used with SGD terminal emulators . |
| keystore | A database of cryptographic keys. A keystore can contain both public keys and private keys . |
| kiosk mode | SGD display mode where an application is displayed full-screen. |



L

| | |
|------------------------------|--|
| LDAP | Lightweight Directory Access Protocol. |
| LDAP directory | A set of LDAP objects organized in a logical and hierarchical manner. |
| LDAP search filter | An RFC2254-compliant search filter, used to select objects in an LDAP directory . |
| LDAP URL | An RFC1959-compliant URL, used to select objects in an LDAP directory . |
| LDAPS | Lightweight Directory Access Protocol over SSL . Used for secure connections to an LDAP directory. |
| load balancing groups | The mechanism that delivers the best possible user experience by choosing SGD servers and application servers linked by a fast network where possible. |
| locale | A set of parameters that defines the user's language, country, and other location-specific preferences. |
| local repository | A store containing information about users, applications, webtops, and application servers. Stored on the primary SGD server and replicated to other SGD servers in the array. Corresponds to the <code>_ens</code> namespace in the SGD datastore. Can be managed using the Administration Console or the <code>tarantella</code> commands. |
| log filter | A string used to configure error reporting to the SGD log files. |
| login script | A script that runs on the SGD server when a user starts an application. Connects to the application server, supplies authentication credentials for that server, and starts the application. |
| LPD | Line Printer Daemon. A printing protocol used to provide print server functions to a UNIX or Linux platform system. Also known as LPR . |
| LPR | Line Printer Remote. See also LPD . |

M

| | |
|----------------------------|--|
| member | A constituent of a group or a role. In SGD, Group objects and Role objects contain one or more member objects. These are usually Application objects, User Profile objects, or Application Server objects. |
| multiple assignment | In the Administration Console, an object link that has both direct assignment and indirect assignment sources. See also Assignment Type . |
| MUPP | MultiplePlexing Protocol. |
| My Desktop | A feature of SGD that enables users to log in and display a full-screen desktop, without displaying an SGD webtop. |

N

| | |
|---------------------|---|
| NetBIOS name | An identifier for a computer running Microsoft Windows. The NetBIOS name can be specified when Windows networking is installed or configured on the computer. |
| NFS | Network File System. |
| NIC | Network Interface Card. |
| NTP | Network Time Protocol. |

O

| | |
|---------------------------------|---|
| object | A self-contained entity, defined by a number of attributes and values. SGD objects have different types, such as X Application or Character Application. The available attributes for each type are defined by a schema. |
| Organization object | An SGD object used to represent the top level of an organizational hierarchy. Organization objects can contain OU or User Profile objects. Organization objects have an <code>o=</code> naming attribute. |
| organizational hierarchy | The collection of objects in the SGD datastore , descending from one or more Organization or Domain Component objects. Represents the collection of people, application servers, and applications within an organization. |

| | |
|-----------------------------------|--|
| Organizational Unit object | An SGD object used to distinguish different departments, sites, or teams in an organizational hierarchy. Organizational Unit (OU) objects can be contained in an Organization or Domain Component object. Organizational Unit objects have an <code>ou=</code> naming attribute. |
| OSS | Open Sound System. A standard interface for audio recording and reproduction in UNIX platform operating systems |
| OU | See Organizational Unit object . |

P

| | |
|-----------------------|---|
| PAM | Pluggable Authentication Modules. |
| passcode | In SecurID authentication, the combination of the PIN and the tokencode . |
| password cache | Short form of application server password cache. |
| PCL | Printer Command Language. |
| PCM | Pulse Code Modulation. |
| PC/SC | Personal Computer/Smart Card. A standard for interoperability of PCs, smart card readers, and smart cards. |
| PDF | Portable Document Format. |
| PDF printing | An SGD feature available for client devices with Adobe Reader software installed. Enables users to print to a PDF printer from their application, which either displays the file or prints using the Adobe Reader program on their client device. |
| peer DNS name | The name by which an SGD server is known to other SGD servers in the same array. |
| PEM | Privacy-Enhanced Mail. Protocol based on public key cryptography . |
| PIN | Code supplied to a SecurID device using a key pad. Combined with a tokencode to form a passcode . |
| PKCS | Public Key Cryptography Standards. Specifications produced by RSA Laboratories for public key cryptography . |
| PKI | Public Key Infrastructure. A security infrastructure based on public key cryptography . |
| primary server | The SGD server that acts as the authoritative source for global information, and maintains the definitive copy of the SGD datastore . |

| | |
|--------------------------------|---|
| print queue | A number of print jobs placed in a storage area on disk. |
| private key | In public key cryptography , a key that is only know by the recipient of a message. The private key can be used to decrypt messages and to create digital signatures . |
| Protocol Engine | An SGD software component that runs on an SGD server. Protocol Engines emulate native protocols such as X11 and RDP and communicate with application servers, sending display data using AIP to Display Engines on client devices. See also application session . |
| proxy server | A server that acts as an intermediary between a client device and the Internet. The proxy server can provide access control and web request caching services. |
| public key | In public key cryptography , a key that can be distributed to anyone. The public key can be used to encrypt messages and to verify digital signatures . |
| public key cryptography | A cryptographic system using a pair of keys, a public key and a private key . The public key is used to encrypt messages and the private key is used to decrypt messages. |

R

| | |
|------------------------------------|--|
| RAM | Random access memory. |
| RDN | See relative distinguished name . |
| Remote Desktop Services | Microsoft Windows software that enables client devices to run applications and access data on a networked Windows server. From Windows Server 2008 R2, Remote Desktop Services is the name for Terminal Services . |
| RDP | Remote Desktop Protocol. Protocol that allows a user to connect to a computer running Windows Terminal Services . |
| RDP printing | Another name for SGD printing from application servers using Windows Terminal Services . |
| relative distinguished name | In an LDAP directory , the part of a distinguished name that uniquely identifies a child entry for a common parent entry. |
| registry | Microsoft Windows registry. On Windows client devices, a database of settings for the operating system. |
| repository | A store containing user information. |

| | |
|-------------------------|--|
| resumability | The attribute of an application session that controls its lifetime. Defined on a per-application basis by an SGD Administrator, as either never resumable, resumable during the user session, or always resumable. See also resume and suspend . |
| resume | To redisplay an application session that has been suspended. See also suspend . |
| RGB value | Defines a color in the RGB color model. The amount of red, green, and blue in the color are indicated by a value from 0 to 255. |
| roaming profiles | A feature of SGD that provides Microsoft Windows users with the same working environment, no matter which Microsoft Windows computer they use. |
| Role object | An object that defines the members and applications associated with a particular role in SGD. Currently, only one role is available, <i>Global Administrators</i> . This role defines the SGD Administrators . |
| root certificate | A self-signed certificate issued by a root level Certificate Authority . |

S

| | |
|---|--|
| Samba | Software that enables a UNIX or Linux platform server to act as a file server for Windows client devices. Uses a variant of the SMB file sharing protocol. |
| SCF | Solaris Card Framework. |
| seamless windows | An SGD window display mode used for Windows applications. Causes an application's windows to behave in the same way as an application running on a Microsoft Windows application server, regardless of the user's desktop environment. Requires the SGD Enhancement Module . |
| secondary server | An array member that is not the primary server . The primary server replicates information to secondary servers. |
| secure connection | A connection between client device and SGD server that uses SSL to protect AIP traffic from eavesdropping, tampering, and forgery. Not related to HTTPS traffic. |
| secure intra-array communication | Secure, encrypted, communication between SGD array members. Uses SSL . |
| SecurID | An authentication mechanism developed by RSA Security to authenticate a user to a network resource. |
| self-signed certificate | An SSL certificate signed by the person who created it. |

| | |
|----------------------------------|--|
| serial port | A physical interface on a computer through which information is transferred one bit at a time. |
| server affinity | Where possible, SGD runs an application on the same application server as the one used to run the previous application for the user. See also application load balancing . |
| session grabbing | The situation where a user logs in to an SGD server, but they already have a user session on another SGD server. The user session is transferred to the new SGD server and the old session ends. |
| SGD | Oracle Secure Global Desktop software. |
| SGD Administrator | An SGD user with permission to configure SGD settings and create and edit SGD objects, either using the Administration Console or the tarantella commands . |
| SGD Client | An SGD component that can be installed on client devices. The SGD Client maintains communication with the SGD server and is required to run applications. |
| SGD Client Helper | A Java applet that downloads the SGD Client . |
| SGD server | A collection of SGD software components that together provide SGD functionality. |
| SGD Web Server | A pre-built web server installed and configured along with the SGD server. Contains Apache, <code>mod_ssl</code> for HTTPS support, and Tomcat for Java Servlet and JSP support. |
| SGD web services | A collection of APIs that allow developers to build their own applications to work with SGD. The APIs can be used to authenticate users, launch applications, and interact with the SGD datastore. |
| SHA | Secure Hash Algorithm. In cryptography, an algorithm that computes a fixed-length representation of a message, called a message digest. |
| shadowing | When an SGD Administrator displays and interacts with a user's application at the same time as the user. |
| SKID | Secret Key Identification. An authentication protocol where a shared secret is used to authenticate a connection. |
| smart card | A plastic card, about the size of a credit card, with an embedded microchip that can be loaded with data. |
| smart card authentication | Authentication to a Windows application server by means of user data contained on a smart card. |
| SMB | Server Message Block. |
| SOAP | Simple Object Access Protocol. A protocol for sending XML messages over computer networks using HTTP. |

| | |
|----------------------------------|---|
| SOCKS | A protocol used by proxy servers to handle TCP connection requests from client devices inside a firewall. |
| SSH | Secure Shell. A secure network protocol for data exchange between two computers. |
| SSL | Secure Sockets Layer. A cryptographic protocol designed for secure Internet communications. |
| SSL certificate | A digital passport that establishes credentials on the web. In SGD, allows client devices to trust the identity of an SGD server. |
| standard connection | A connection between a client device and an SGD server that is not secured. This is the default connection mode when using SGD. |
| subject alternative names | Alternative DNS name , other than the hostname, specified for an SGD server on an SSL certificate . |
| suspend | To pause an application session. A suspended application is not closed down, it can be resumed. See also resume . |
| system authentication | A component of the SGD server that authenticates users against an external authentication service, such as a Windows domain or an LDAP directory, and determines a user's SGD user identity and user profile. |

T

| | |
|----------------------------------|--|
| tarantella command | An SGD administration tool available from the command line. Used to control the SGD server and make configuration changes. |
| Tarantella System Objects | The Organization object in the SGD datastore that contains objects essential for smooth running and maintenance of SGD. |
| Tcl | Tool Command Language. A scripting language developed by John Ousterhout. The SGD login scripts include some Tcl functions. |
| TCP | Transmission Control Protocol. |
| TCP/IP | Transmission Control Protocol/Internet Protocol. |
| terminal emulator | A program that runs on a graphical user interface and emulates a "dumb" video terminal. SGD includes terminal emulators for SCO Console, Wyse 60, and VT420 terminals. |
| Terminal Services | Microsoft Windows software that enables client devices to run applications and access data on a networked Windows server. From Windows Server 2008 R2, Terminal Service is renamed Remote Desktop Services . |

| | |
|-----------------------------------|--|
| third-party authentication | A component of the SGD server that trusts authentication information supplied by a third party and uses that information to automatically authenticate the user as an SGD user, allocating a user identity and a user profile. |
| token cache | A store for tokens used by the authentication token authentication mechanism. |
| tokencode | A random number generated by a SecurID device. Combined with a PIN to form a passcode . |
| ttaserv, ttasys | Users and a group (<i>ttaserv</i>) that must be set up on a system before SGD can be installed. These users and group own some SGD files and processes after installation. |

U

| | |
|------------------------------------|---|
| UCX | Ultrix Communications Extensions. |
| UDP | User Datagram Protocol. |
| UNC | Universal Naming Convention. |
| Unicode | A standard for universal character encoding. Provides the basis for processing, storage, and interchange of text data in any language. |
| URL | Uniform Resource Locator. |
| user identity | The SGD concept of who a user is. A user identity can belong to one of a number of different namespaces. User identities are allocated by authentication mechanisms. The user identity can be the same as the user profile in some cases. |
| user principal name | In Active Directory , the required format for user names. The user principal name is in email address format, for example, <code>indigojones@indigo.insurance.com</code> . |
| User Profile object | An SGD object that represents a user in an organization. Can be used to give a user access to applications. User Profile objects can have a <code>cn=</code> (common name), a <code>uid=</code> (user identification), or a <code>mail=</code> (mail address) naming attribute. |
| user session | Begins when a user logs in to SGD, and ends when the user logs out. Information about a user session is stored in memory by the SGD server. |
| user session load balancing | The mechanism that determines which SGD server in the array a user logs in to to display their webtop . |
| UTC | Coordinated Universal Time. |

V

| | |
|------------------------|---|
| virtual hosting | Hosting of multiple web servers on the same computer. Each web server has a different DNS name . |
| VMS | Virtual Memory System. Operating system originally developed for use on the VAX and Alpha family of computers from DEC. |

W

| | |
|-----------------------------------|--|
| WAN | Wide Area Network. |
| WAR | Web Application Archive. |
| webtop | A web page where users can run applications using SGD, view documents, and manage print jobs. Can be accessed using a web browser or the SGD Client. |
| webtop content | The collection of applications and documents that appear on a user's webtop . |
| webtop inheritance | The ability to define webtop content implicitly. Content is usually inherited from the parent object, but other objects can also be used. |
| webtop link | A hyperlink on an SGD webtop that the user clicks to starts an application. |
| Webtop mode | The mode of operation of SGD where you use a browser to display the SGD webtop . |
| Windows Application object | An SGD object that represents a Microsoft Windows graphical application. Windows Application objects have a <code>cn=</code> naming attribute. |
| Windows domain | A logical group of computers running the Windows operating system. |
| Windows domain controller | A server in a Windows domain that hosts the Active Directory . The domain controller handles authentication of users and administration tasks. |
| Windows protocol | In SGD, the protocol used to connect to an application server hosting a Microsoft Windows application. |
| WINS | Windows Internet Name Service. |

X

| | |
|-----------------------------|---|
| X.509 certificate | See SSL certificate . |
| X11 forwarding | The process of forwarding, or tunneling, the windows of a remotely started X application to a client desktop. |
| X11 protocol | Display protocol used for the X Window System . |
| X Application object | An SGD object that represents an X11 graphical application. X Application objects have a <code>cn=</code> naming attribute. See also X11 protocol . |
| X authorization | Access control mechanisms that control whether a client application can connect to an X server. |
| X Window System | A distributed window system for UNIX platform operating systems, based on the X11 protocol . Also called X11, or X Windows. |

Z

| | |
|--------------|--|
| zones | A feature of Solaris 10 OS that enables multiple virtual operating systems to be deployed on a single Solaris OS server. |
|--------------|--|

Index

Numerics

- 3270 applications
 - configuring, 202
 - objects, creating, 712
 - settings for, 526
- 5250 applications
 - configuring, 202
 - objects, creating, 717
 - settings for, 528

A

- Active Directory
 - authenticating users
 - about, 52
 - enabling, 60
 - blacklists, 97
 - domain lists, 99
 - enabling, 440, 445
 - Kerberos
 - configuring, 56
 - enabling, 450
 - LDAP discovery, 100
 - LDAP discovery timeout, 90
 - LDAP operation timeout, 99
 - logging in
 - domain name, 446, 450
 - lookup cache timeout, 99
 - password, 449
 - password expiry, 95
 - ports, 17
 - search only global catalog, 97
 - securing connections, 450
 - server URL, 447, 448
 - service object, 90
 - creating, 92
 - sites, 96
 - SSL connections, 56
 - suffix mappings, 98
 - tuning authentication, 86
 - user name, 449
 - whitelists, 97
- Active Directory containers
 - objects, creating, 727
- Active Directory users
 - restricting logins for, 87
- Administration Console
 - browsers, supported, 381
 - controlling access to, 385
 - deployment descriptor settings, 383
 - limitations of, 382
 - starting, 382
- Administrators
 - adding, 122
 - deleting, 122
 - webtops for
 - adding applications to, 793, 817
 - removing applications from, 797
- Advanced Load Management
 - description of, 371
 - troubleshooting, 423
- AIP commands
 - compressing, 578
 - optimizing, 579
- ALSA audio driver, 279
- anonymous users

- description of, 390
 - enabling, 442
 - logging in, 61
- Apache
 - log files, 406
 - SGD web server and, 380
- application objects
 - creating, 730
- application server objects
 - creating using commands, 733
- application servers
 - AS/400, 639
 - assigning to applications, 604, 703
 - authenticating users, 592
 - available applications, 602
 - caching passwords for, 762
 - connecting to, 581
 - connections, securing, 2
 - DNS name, 545
 - enabling, 554
 - IP address, 545
 - load balancing properties, 375
 - login scripts and, 863
 - mainframe, 639
 - Microsoft Windows, printing from, 225
 - removing from load balancing pool, 755
 - securing connections to, 179
 - server affinity, 366
 - settings for, 530
 - UNIX or Linux platforms, printing from, 228
- application sessions
 - definition of, 389
 - displaying information for, 695
 - ending, 521, 640, 698
 - instances, maximum number of, 630
 - listing, 693
 - load balancing, 361
 - managing, 389, 521, 552, 692
 - shadowing, 196, 214, 521, 696
 - sharing system resources, 642
 - showing details of, 521
 - suspending, 697
- applications
 - assigning to application servers, 124, 602
 - assigning to LDAP groups, 129
 - assigning to LDAP users, 128
 - assigning to users, 124, 556, 704
 - assigning using LDAP searches, 130
 - assignments, inheriting, 607
 - display windows, configuring, 665
 - load balancing
 - algorithm, 548
 - description of, 362
 - tuning, 372
 - resuming, 549, 551
 - starting, keeping connection open, 608
 - suspending, 549
- array connections
 - securing, 3, 336
- array failover
 - backup primaries
 - adding, 677
 - changing position of, 680
 - listing, 683
 - removing, 685
 - definition of, 338
 - enabling, 349, 351, 483
 - examples of, 341
 - logging, 422
 - ping attempts, number of, 484
 - ping interval, 484
 - primary find attempts, number of, 486
 - primary find interval, 485
 - restoring the original state, 678
 - return of primary, 486
 - secondary servers, promoting, 338
 - status information, 422
- array routes
 - configuring, 11
 - definition of, 11
- arrays
 - adding a server, 338, 343, 494, 681
 - changing the primary server, 338, 343, 494
 - clock synchronization and, 334
 - configuring, using the Administration Console, 338, 343
 - copying data between servers, 334
 - creating, 675
 - definition of, 333, 493
 - evaluation licenses and, 334
 - listing members, 683
 - managing
 - Administration Console, using, 383
 - peer DNS names, 335
 - promoting a secondary, 684
 - removing a server, 338, 343, 494, 679

- removing array state information, 352
- returning to original state, 678
- settings, configuring, 688
- status information, 830
- synchronizing, 334, 461, 503
- assigned applications
 - Administration Console, using, 556
 - inheriting, 607
 - LDAP, using, 135
 - using LDAP, 127
- assignments
 - definition of, 107
 - for applications, 124
 - LDAP users, 127, 135
 - local, 125
 - viewing, using Administration Console, 132
- attributes, definition of, 107
- audio
 - client devices, configuring, 280
 - data compression, enabling, 519
 - enabling
 - UNIX platforms, 469
 - Windows, 467
 - Enhancement Module and, 277
 - ESD, using, 280
 - logging, enabling for X applications, 281
 - sample rate, 468, 469
 - services, enabling, 279
 - troubleshooting, for applications, 281
 - Windows applications, configuring, 276
 - X applications, configuring, 276
 - X applications, device selection, 563
- authentication dialog, smart cards and, 293
- authentication tokens
 - deleting, 325
 - disabling, 325
 - enabling, 323, 434
 - enabling log ins, 441
 - managing, 490, 647
 - using for automatic logins, 323
 - viewing, 325
- Authentication Wizard, 432
- automatic logins
 - enabling, 323
 - setting log filters for, 327

B

- backing up, installation files, 414
- backup primaries list
 - adding an entry, 677
 - definition of, 338
 - displaying, 683
 - reordering entries, 680
- backup primaries table, 487
- bandwidth, restricting, 427
- batch scripts
 - managing objects with, 761
 - objects, creating, 117
- billing
 - enabling, 482
 - generating a report, 784
- blacklists
 - Active Directory, 97
- browsers
 - Administration Console and, 381
 - security warnings, 33

C

- CA certificates
 - array connections and, 336
 - chains, installing, 25, 28
 - fingerprint, displaying, 812
 - importing, 408
 - installing, 807
 - intra-array communications, for, 813
 - removing, 807
 - truststore, location of, 408
- CALs
 - description of, 144
 - managing, 839
- CAs, supported, 19
- CDE
 - desktop sessions, configuring, 198
 - SSH issues, workaround, 198
- CDM
 - application servers, configuring, 259, 262
 - configuring, 259
 - diagnostics, 275
 - enabling, 262, 465, 466
 - examples of, 266
 - guest users, 274
 - logging, enabling, 275
 - managing, 567

- services
 - stopping, 834
- services, starting, 828
- troubleshooting, 267
- chaining
 - SSL certificates, 25, 28
- character applications
 - colors, 576
 - configuring, 159
 - objects, creating, 722
 - settings for, 531
 - settings, typical, 159
- cipher suites
 - client connections, for, 41
 - intra-array connections, for, 347
- client certificates
 - installing, 409
 - keystore, location of, 409
- client connections
 - bandwidth, maximum, 565
 - filters, 582
 - IP filters, 3, 43
 - keepalive messages, timeouts, 459
 - minimum packet size, 516
 - ports, 13
 - proxy settings, 8
 - securing, 2, 18, 582
 - security warnings, 33
 - SSL accelerators, using, 40
 - using data compression, 515
- client devices, settings, 465
- client drives
 - access to, configuring, 259, 264, 265
 - enabling, 465, 466
 - enabling access to, 467, 567
- client overrides
 - configuring, 174
 - definition, 174
- client printers
 - about, 224
 - configuration, 237, 241
- client profiles
 - definition of, 311
 - editing, enabling for users, 572
 - enabling users to edit, 312
 - locations of, 317
 - managing, 312
 - roaming users and, 319
 - settings for, 315
 - settings for automatic logins, 328
 - user editing of, enabling, 473
- clipboard
 - enabling, 471
 - enabling access to, 585
 - securing, 587
 - security levels, 288, 472
 - examples of, 289
 - troubleshooting, 291
- clock synchronization
 - arrays, 334
 - Kerberos, 55
- clocks
 - synchronizing, 420
- color maps, terminal emulators and, 167
- colors
 - mapping to RGB values, 508
 - number of, 574
 - quality, for X applications, 576
 - X applications, 216
 - X displays, 657
- COM ports, mapping, 298
- commands
 - about, 672
 - help text for, 699
- connection types
 - configuring, 43
 - enabling, 43, 480
- copy and paste, 287
 - access control, 288
 - applications and, 587
 - configuration tips, 290
 - enabling, 471
 - securing, 472
 - troubleshooting, 291
 - users, enabling for, 585
- CSRs
 - displaying information about, 801
 - generating, 803
 - SGD servers, generating for, 20
- CUPS, 232

D

- datastore
 - definition of, 386

- namespaces and, 386
- desktop sessions
 - JDS, reducing bandwidth, 195
 - Windows, improving performance of, 194
- detaching, SGD servers from an array, 338, 343, 679
- diagnostics, client drives, for Windows
 - applications, 275
- directory servers
 - versions, supported, 65
- Directory Services Integration (DSI), definition of, 127
- display size
 - height, 509
 - width, 509
- display windows
 - clipping, 218, 509
 - closing, application behavior and, 640, 654
 - full screen, 660
 - height, 661
 - scaling, 663
 - types of, 665
 - width, 664
- DNS names
 - external, 5
 - for an SGD host, 3
 - peer, 3, 6
- DNS names, for SGD servers, 495
- documents, 196
 - displaying, 668
 - objects, creating, 728
 - settings for, 536
 - URLs for, 649
- domain components
 - objects, creating, 727
- domain controllers, configuring, 85
- domain name, for Windows domain
 - authentication, 444
- domains
 - Active Directory, 99
- drive letters
 - remapping, 262
- dynamic application servers
 - client overrides, 174
 - configuring, 169
 - definition, 169
 - virtual server brokers, 169

- dynamic applications
 - application mapping, 172
 - client overrides, 174
 - configuring, 172
 - definition, 172
- dynamic launch
 - client overrides, 174

E

- Enhancement Module
 - audio module, installing, 277
- environment variables
 - applications and, 594
 - TTASSHCLIENT, 179
- error messages
 - logging, 482
 - login scripts, for, 878

F

- fingerprints, for CA certificates, 812
- firewall forwarding
 - enabling, 25, 28
 - See firewall traversal, 24
- firewall traversal
 - definition of, 24
 - enabling, 810
 - redirecting non-SGD web requests, 499
 - SGD web server, configuring, 24
 - troubleshooting, 428
- firewalls
 - ports, for SGD, 13
 - using, 13
- fonts
 - location of, 507
 - PDF printing, troubleshooting, 257
 - size of, 598, 626
 - typeface, 597
 - X applications
 - configuring, 156
 - troubleshooting, 215
- full screen displays
 - configuring, 665
 - escape key, 658
 - escaping from, 612

G

- Ghostscript, 224, 233

- global catalog
 - Active Directory, 97
- global settings, 688
- graphics
 - animated, problems with, 215
 - application icons, for webtop, 606
 - color quality, 576
 - display updates, improving performance and, 590
 - high-quality, enabling, 601
 - interlacing, for low bandwidth connections, 608
 - number of colors, 574
 - optimizing, for performance, 601
- groups
 - adding members to, 622
 - application servers, of, 536
 - applications, of, 536
 - deleting members from, 622
 - members, adding, 707
 - objects, creating, 731
 - removing members from, 759
- guest users, 105, 616
 - CDM and, 274
 - description of, 390

H

- HTTPS
 - certificate warning messages, 33
 - SGD web server, using, 18

I

- ICA
 - enabling, for Windows applications, 642
 - printing, configuration of, 228
- idle timeouts, for user sessions, 388
- input methods, enabling, 188
- installation directory
 - backing up, 414
 - contents, 411
 - files and directories, 414
 - location, 411
 - restoring, 414
- Integrated mode
 - authentication tokens
 - enabling, 434
 - managing, 836
 - definition of, 301, 320

- logging in, enabling, 441
- resuming applications, 329
- SGD Client and, 322
- suspending applications, 329
- intra-array communication
 - ports, 14
- intra-array communications, 335
 - cipher suites, selecting, 347
 - securing, 3, 813
 - SSL certificates for, 336

J

- Java Plugin tool
 - security warnings, 33
- JDS desktop sessions, reducing bandwidth for, 195
- joining, SGD servers to an array, 338, 343, 681
- JVMs, RAM usage, 502

K

- Kerberos
 - authenticating users, 52
 - configuration files, 56
 - configuration, resetting, 686
 - synchronizing clocks, 55
- key pairs, generating, 20, 803
- keyboard maps
 - default, 509
 - euro character, 595
 - locking, 611
 - Sun keyboards, 218
 - terminal emulators, 161, 573, 610
 - users, 610
 - Windows applications, configuring, 147
 - X applications, configuring, 158
- keyboard shortcuts
 - full screen displays, enabling, 658
 - sending to remote session, 658
 - Terminal Services, 147
- keystrokes
 - sending to remote session, 147
 - VT420, escape sequence, 595
- kiosk mode, 665
 - exiting from, 147
 - pull-down headers, enabling, 612
 - scaling the display window, 215

L

languages

- changing, for webtop, 329
- input methods, enabling, 188
- keyboard maps for, 509
- login script prompts, 635
- system prompts for, 189

Launch menu, starting SGD from, 320

LDAP

- assigning applications to users, 558
- assigning objects with, 127
- authenticating users
 - description of, 63
 - enabling, 67
- data cache, flushing, 686
- discovery timeout, 90
- enabling, 440, 445
- groups
 - assigning to applications, 129
 - searching, 132
- lookup cache timeout, 99, 100
- mirroring data
 - description of, 118
 - examples of, 119
- operation timeout, 99
- password, 449
- password expiry, 95
- password update, 96
- ports, 17
- searches
 - filters, using, 130
 - for assigning applications, 130
 - for assigning users, 558
 - tuning of, 132
 - URLs, using, 130
- server URL, 447, 448
- service object, 90
 - creating, 93
- tuning authentication, 86
- URLs, specifying, 118
- user name, 449

LDAP users

- assigning to applications, 127
- logging in, 63
- restricting logins for, 87
- user profiles, matching, 437, 438, 439
- webtops, 127

licenses

CALs, managing, 839

Terminal Services, 144

load balancing

- application servers, removing, 703, 755
- application sessions, 361, 463
- applications, 362, 464
 - Advanced Load Management, 371
 - algorithm for, 548
 - application servers, selecting, 362, 364
 - geographical groups, 614
 - load balancing groups, 363
 - methods, 363
 - properties file, 375
 - tuning, 372
- definition of, 352
- geographical groups, 503
- intelligent array routing, 614
- load balancing JSP, using, 353
- server affinity, 366
- SGD Gateway, using, 353
- troubleshooting, 423
- user sessions, 353
 - external load balancer, using, 356
 - hardware load balancers, using, 356
 - load balancing JSP, using, 354
 - My Desktop, configuring for, 359
 - round robin DNS, using, 356

local repository

- assignments between objects, 125
- definition of, 107
- mirroring LDAP data, 118
- user profiles, matching, 436

log files

- archiving, 675
- protocol engines, 786
- searching, 781

log filters, 781

- configuring, 482
- examples of, 397, 401, 403
- for auditing, 398
- for authentication troubleshooting, 102
- for automatic logins, 327
- for protocol engines, 402
- for SGD servers, configuring, 391
- for SSL, 38

logging

- Apache, 406
- audio, for X applications, 281

- client drives, 275
- configuring of, 481
- protocol engines, 402
- SGD Client, 407
- SGD servers, configuring, 391
- SGD web server, 406
- SSL processes, 38
- system events, 398
- Tomcat, 406

logging in

- authenticating users, 432
- authentication mechanisms, 49
- automatically, using authentication tokens, 323
- caching SGD passwords, 435
- connection types, using, 480
- disabled logins, warning message for, 497
- disabling, for users, 615
- enabling
 - for Linux users, 80
 - for SGD servers, 496
 - for UNIX users, 80
 - for users, 615
- external authentication, using, 72
- first time, warnings, 35
- guest users, 105, 616
- number of attempts, 103
- security warnings, 34
- smart cards, using, 293
- user name, 617
- users unable to log in, 104

logging out, 852

login scripts

- applications, specifying for, 618, 856
- commands for, 858
- description of, 184
- error messages, 878
- language used for prompts, 635
- location of, 514
- procedures for, 858
- timeouts, 874
- variables for, 866
- VMS applications, configuring, 201

M

mouse devices

- middle button, emulating, 625
- three-button, 627

- multiple monitors, configuring applications for, 191

My Desktop

- configuring, 175
- exit problems, 212
- load balancing, configuring, 359

N

- namespaces, used in datastore, 386

- naming conventions, for objects, 110, 628

- NFS shares, and CDM, 260

O

object names

- conventions for, 116

- object types, 110

objects

- 3270 applications, 526

- 5250 applications, 528

- Active Directory containers, 535

- application servers, 530

- assignments and, 107

- attributes

- definition of, 107

- editing, 709

- listing, 710

- changing name of, 760

- character applications, 531

- containers, listing contents of, 711

- creating, 110, 700

- definition of, 107

- deleting, 700, 708

- description of, 579

- documents, 536

- domain components, 535

- editing, 700

- groups, 536

- LDAP schema and, 107

- managing

- using the Administration Console, 108

- using the command line, 700

- naming conventions for, 110, 628

- organizational units, 534

- organizations, 107, 533

- user profiles, 537

- Windows applications, 540

- X applications, 542

- organization objects, creating, 736

- organizational hierarchies

- definition of, 107

- designing, 115
- for objects, 108
- populating, using scripts, 117
- organizational units, settings for, 534
- organizations
 - settings for, 533
- OU objects, creating, 738

P

- password cache
 - adding new entries, 769
 - changing entries, 765
 - definition of, 185
 - deleting entries, 763
 - encrypting entries, 186
 - generating a new key, 478
 - listing entries, 767
 - managing, 185, 488, 631, 762
 - missing passwords, authentication policy for, 632
 - namespaces used, 488
 - SGD passwords, saving, 435
 - Windows domain name and, 187
- password expiry
 - Active Directory, 95
 - LDAP, 95
- passwords
 - expired, 50, 452
 - SGD, saving in password cache, 435
- PDF printing
 - configuring, 225
 - default printer, 619, 620
 - definition of, 224
 - fonts, troubleshooting, 257
- printer drivers
 - default, 226
 - enabling, 477
- SGD printers, 647, 648
 - default, 475, 477
 - enabling, 475, 476
 - printer drivers, 477
- Universal PDF printer, definition of, 224
- Universal PDF viewer, definition of, 224
- Windows applications
 - default printer, 619, 620
 - enabling, 647, 648
 - printer driver, configuring, 633
- Windows clients, for, 237

- PDF viewers, supported, 241
- Pluggable Authentication Modules (PAM), 82
- ports
 - Active Directory, 17
 - application servers, connections, 15
 - client connections, 13, 457, 458
 - intra-array communication, 14
 - intra-array communications, 335
 - LDAP, 17
 - opening, for SGD, 13
 - secure connections, 458
 - SecurID, 17
 - terminal emulators, 639
- PostScript printers, PDF printing and, 633
- PostScript, converting to PCL, 235
- primary servers
 - changing, 338, 343
 - definition of, 493
- print jobs
 - canceling, 772
 - format, converting, 235
 - life time of, 245
 - listing, 773
 - managing
 - commands for, 244
 - using the webtop, 245
 - moving to another SGD server, 774
- printer drivers
 - default, for PDF printing, 226
 - LaserJet 8500, 633
 - name mapping, 238
 - PDF printing, 475, 476, 477
 - Windows application servers and, 225
- printer queues
 - disabling, 256
 - installing, 228
- Printer-Direct printing
 - client printers, defining, 242
 - configuring, 225
 - definition of, 224
 - Windows clients, for, 238
- printing
 - _Default printer, 258
 - Citrix ICA, configuring, 228
 - client devices, enabling for, 474
 - configuring, 225
 - CUPS, using, 232

- data compression, using, 517
- distributed printing, 224
- lp command, 232
- lpr command, 232
- Microsoft Windows clients, 237
- minimum packet size, 517
- PCL, 224
- RDP, configuring, 225
- remote print requests, 234
- text-only, 224
- timeouts, 479
- troubleshooting, 246
- printing services
 - pausing, 775
 - resuming, 776
 - starting, 777
 - status reporting, 778
 - stopping, 779
- private keys, decrypting, 808
- Profile Editor, 312
- protocol engines
 - audio, data compression, 519
 - channel
 - data compression, 515
 - minimum packet size, 516
 - timeouts, 516
 - character
 - command options, 506
 - timeouts, 505
 - tuning, 505
 - configuring, 504
 - error logs, displaying, 786
 - execution
 - command options, 514
 - timeouts, 513
 - tuning, 512
 - logging, 402
 - print
 - data compression, 517
 - minimum packet size, 517
 - timeouts, 518
 - smart card, data compression, 519
- X
 - command options, 512
 - connection timeouts, 510
 - default keyboard map, 509
 - display size, 509
 - dpi setting, 507

- font locations, 507
- RGB colors, mapping, 508
- timeouts, 511
- tuning, 511

- proxy servers
 - application startup issues, 209
 - array routes, 11
 - automatic configuration scripts, 10
 - client configuration, 9
 - exception lists, 10
 - for client connections, 8
 - supported, 8

R

RDP

- audio, configuring, 277
- client printers, configuring, 569, 570
- command-line arguments, 554
- enabling, for Windows applications, 642
- printer preferences, 255
- printing, configuration of, 225, 474
- Remote Desktop, using, 145
- ttatsc command, 149

- Remote Desktop, configuring, 145

- resuming applications, 551
 - application sessions and, 387
 - settings, 549
 - timeouts, 459, 460

- roaming users
 - client profile locations, 319
 - installing the SGD Client, 303

roles

- adding users to, 794
- assigned applications
 - adding, 793
 - listing, 795
 - removing, 797
- listing, 795
- members of
 - listing, 796
 - removing, 798
- SGD Administrators and, 122, 791

S

scripts

- print jobs, converting format of, 235
- printer queues, installing, 230

- seamless windows

- Borland applications and, 645
 - configuring, 665
 - for Windows applications, 146
- secondary servers
 - definition of, 493
 - detaching from array, 679
 - joining to array, 681
 - promoting to primary, 684
- secure connections
 - arrays, enabling, 336
 - cipher suites, 347
 - client connections, cipher suites for, 41
 - clients, enabling, 43
 - disabling, 809, 815
 - enabling, 498, 810, 814
 - port 443, using, 24
- SecurID
 - Agent Hosts, configuring, 71
 - applications, authenticating, 189
 - logging in, enabling, 72, 442
 - ports, 17
 - setting up, 70
 - users, authenticating, 69
 - versions, supported, 70
- security services
 - about, 2
 - disabling, 809
 - enabling, 18, 25, 28, 810
 - starting, 814
 - stopping, 815
- serial ports
 - accessing from Windows applications, 298
 - client access, enabling, 637
 - configuring, for client devices, 299
 - enabling access to, 470
- server affinity, 366
- service objects, 90
 - Active Directory
 - creating, 92
 - blacklists, 97
 - creating, 822
 - deleting, 817
 - domain lists, 99
 - editing, 817
 - LDAP
 - creating, 93
 - listing, 821
 - lookup cache timeout, 99
 - operation timeout, 99
 - password expiry, 95
 - password update, 96
 - search only global catalog, 97
 - sites, 96
 - suffix mappings, 98
 - whitelists, 97
- setup, after installing SGD, 826
- SGD Administrators
 - roles and, 791
 - third-party authentication, allowing, 77
- SGD Client
 - automatic logins, configuring, 322
 - commands
 - examples of, 306
 - web services options, 306
 - configuration settings, 312
 - installing
 - automatically, 303
 - manually, 305
 - logging, configuring, 407
 - modes of operation, 301
 - running without Java technology, 310
- SGD Client Helper, 301
- SGD Gateway
 - description of, 24
 - load balancing and, 353
- SGD servers
 - logging in, enabling, 496
 - performance tuning, 500
 - securing
 - automatically, 25
 - manually, 28
 - securing connections to, 810
 - settings, configuration of, 494, 688
 - starting, 789, 829
 - stopping, 789, 835
 - uptime, 787
- SGD web server
 - components of, 380
 - enhanced security configuration, 380
 - HTTPS connections, using, 18
 - securing connections to, 18
 - starting, 790, 829
 - stopping, 790, 835
- shadowing
 - application sessions, 696
 - low bandwidth connections, 220

- troubleshooting applications with, 214
 - virtual classrooms, creating, 196
 - sites
 - Active Directory, 96
 - smart cards
 - data compression, enabling, 519
 - enabling, 294, 470
 - for authenticating applications, 453
 - readers, client device configuration, 295
 - supported versions, 292
 - troubleshooting, 296
 - Windows applications and, 292
 - SSH
 - application servers, connecting to, 179, 581
 - CDE, using with, 198
 - client keys and, 183
 - command-line arguments, 179, 584
 - configuring, 178
 - environment variables, 179
 - IPv4 mode, enabling, 182
 - securing X displays, 669
 - X Security extension, using, 182
 - X11 forwarding, enabling, 181
 - X11, downgrading to, 181
 - SSL
 - cipher suites, 41, 347
 - client connections, using, 18, 582
 - enabling secure connections, 498
 - log filters for, 38
 - process tuning, 38
 - securing connections with, 2
 - SSL accelerators, enabling, 40, 498
 - SSL certificates
 - alternative names, 19
 - array connections and, 336
 - CAs, supported, 19
 - chaining, 25, 28
 - CSRs, generating, 20
 - displaying information about, 801
 - expired, 23
 - format of, 19
 - installing, self-signed, 813
 - installing, tarantella security certuse, 28, 805
 - installing, tarantella security enable, 25, 810
 - log in messages, avoiding, 35
 - replacing, 23
 - security warnings, 34
 - self-signed, creating, 20, 813
 - SGD web server and, 19
 - Start menu, starting SGD from, 320
 - starting
 - CDM, 828
 - SGD servers, 789, 829
 - SGD services, 827
 - SGD web server, 790, 829
 - starting applications
 - Application Launch dialog, configuring, 456
 - authenticating
 - users, 183
 - using SGD password, 451
 - authentication dialog
 - configuring, 454, 858
 - description of, 183
 - disabling, 212
 - authentication dialog, configuring, 455
 - authentication dialogs
 - configuring, 453
 - command path name, 547
 - command-line arguments, 555
 - environment variables, specifying, 594
 - exits immediately, 207
 - expired passwords, behavior, 452
 - failure of, troubleshooting, 203
 - keeping connection open, 608
 - login scripts, 855
 - progress dialog, configuring, 862
 - proxy server issues, 209
 - SecurID, using, 189
 - smart cards, using, 296, 453
 - timeouts, 874
 - without using a webtop, 190
 - xauth issues, 207
 - status information, for arrays, reporting, 830
 - stopping
 - CDM, 834
 - SGD servers, 789, 835
 - SGD services, 832
 - SGD web server, 790, 835
 - suffix mappings
 - Active Directory, 98
- ## T
- tarantella command, 672
 - tarantella config
 - global settings
 - array-cdm, 467

- tcc command, 306
- Tcl, login scripts and, 858
- telnet
 - application servers, connecting to, 581
 - closing connections, terminal emulator behavior, 580
- terminal emulators, 593
 - answerback message, 546
 - attribute maps, 166
 - code page, 573
 - color maps and, 167
 - configuring, 159
 - keyboard codes, 609
 - keyboard maps, 161
 - keyboard type, 612
 - numeric keypad, 631
 - ports, 639
- Terminal Services
 - CALS, managing, 839
 - desktop sessions, improving performance of, 194
 - features, supported, 140
 - keyboard shortcuts, 147
 - licensing, 144
 - password prompts, 213
 - printer names, changing, 257
 - smart cards, configuring, 293
- terminal windows
 - color of background, 564
 - cursor key behavior, 588
 - cursor style, 588
 - emulation type, 645
 - font size, 598
 - font typeface, 597
 - maximizing, 662
 - menu bar, displaying, 625
 - menus, displaying, 596
 - scrolling, 637
 - size of, 661, 662
 - soft buttons, 591
 - status line, 643
 - style of, 566
 - text colors, 563, 599
 - wrapping lines, 613
- text attributes, mapping, for terminal emulators, 166, 563
- third-party authentication
 - definition of, 72
 - enabling, 76, 435
 - SGD Administrators, allowing log ins, 77
 - trusted users, 77, 846
 - user profiles, default LDAP profile, 438
 - user profiles, default profile, 437
 - user profiles, LDAP repository search, 437
 - user profiles, local repository search, 436
 - user profiles, matching, 73, 439
- third-party software, licenses, 885
- time zones, name mapping, 472
- timeouts
 - channel protocol engines, 516
 - character protocol engines, 505
 - client connections, 459
 - execution protocol engines, 513
 - login scripts, configuring, 874
 - print name mapping, 479
 - print protocol engines, 518
 - resuming applications, 459, 460, 551
 - starting applications, 874
 - user sessions, 462
 - X protocol engines, 510, 511
- TLS
 - cipher suites, 41, 347
 - securing connections with, 2
- token cache
 - deleting entries, 837
 - listing entries, 838
 - managing, 490, 647, 836
- tokens, authentication, managing, 490
- Tomcat
 - deployment descriptor, for Administration Console, 383
 - log files, 406
- trusted users
 - configuring, 846
 - description of, 77
- truststores, locations of, 408
- ttatcc command, 306
- ttatsc command, 149
- tuning
 - applications, starting of, 512
 - audio processes, 518
 - channel protocol engines, 515
 - character protocol engines, 505
 - execution protocol engines, 512
 - file descriptors, 501

- JVM memory size, 502
- print processes, 517
- server requests, 500
- smart card processes, 519
- user sessions, 500
- X protocol engines, 507, 511

U

- uninstalling, SGD, 844
- UNIX users
 - logging in, enabling, 440
 - user profiles, matching, 443, 444
- uptime, for SGD servers, 787
- URLs, directory servers, 447, 448
- user identity, definition of, 48
- user names, ambiguous, 106
- user profiles
 - assigning to applications, 558
 - definition of, 49
 - email address, 593
 - family name, 644
 - objects, creating, 741
 - removing assigned applications, 756
 - search methods, for authentication, 444
 - settings for, 537
- user sessions
 - anonymous users, 390
 - definition of, 387
 - ending, 520, 852
 - guest users, 390
 - idle timeout, 388
 - listing, 851
 - load balancing, 353
 - managing, 387, 520, 652
 - relocating, clock synchronization and, 429
 - showing details of, 520
- users, authenticating, 47

V

- VDI
 - integrating SGD with, 176
- versions, of installed software, displaying, 845
- virtual applications, 730
 - mappings, creating, 705, 757
- Virtual Desktop Infrastructure
 - integrating SGD with, 176
- virtual server brokers, 169

- SGD broker, 170
- User-defined SGD broker, 171
- VDI broker, 172

- VMS applications, configuring, 201

- VT420

- escape keystrokes, 595
 - inquiry response string, 546
 - numeric keypad behavior, 631

W

- WAR files, for Administration Console, 382
- web applications, 196
- web pages
 - displaying, 668
 - URLs for, 649
- web servers
 - SGD web server, 379
- web services
 - SGD Client, command options for, 306
 - starting applications, 190
- webtop
 - adding applications to, 556, 558
 - changing the language, 329
 - definition of, 301
 - DSI, using, 127
 - icons, for applications, 606
 - LDAP users, assigning applications to, 127
- webtops
 - adding applications, 704
 - customizing, 602
- whitelists
 - Active Directory, 97
- window types, 665
- Windows applications
 - audio redirection, 636
 - client drives, accessing, 567
 - client printers, 569
 - command-line arguments, 554
 - configuring, 138
 - console mode, 585
 - cursor settings, 589
 - cursor shadow, 589
 - default printers, changing, 570
 - desktop wallpaper, 591
 - ending session, problems with, 210
 - font smoothing, 599
 - keyboard maps, 149

- keyboard shortcuts
 - remote sessions, enabling, 148
 - supported, 147
- menu animation, 624
- objects, creating, 744
- printer preference caching, 635
- printing, configuration, 474
- running locally, 153
- seamless windows, 146
- serial ports, configuring, 298
- settings, 138, 540
- smart cards
 - configuring, 292
 - enabling, 470
- starting locally, 615
- Terminal Services Group Policies, 144
- themes, 646
- window drag, 600
- working directory, 668

Windows domains

- application server authentication and, 592
- authenticating users, 83
- logging in, enabling, 441

Windows key, enabling, 148

Wyse 60

- inquiry response string, 546
- keyboard codes, 609

X

X applications

- audio, configuring, 279
- client window management (CWM) issues, 218
- configuring, 153
- fonts, configuring, 156
- high color, issues, 216
- objects, creating, 750
- settings, 153, 542

X displays

- background color, 657
- color depth, 574
- font size, 626
- monitor resolution, 507
- securing, 481, 669
- sharing system resources, 642
- Window Managers, 659
- X extensions, supported, 155
- xauth issues, 207

X extensions

- supported, 155
- X Security, and SSH, 182
- X11 downgrade, from SSH, 181
- X11 forwarding, for SSH, 181

xauth

- application start problems, 207
- enabling, 481

